Life Cycle Assessment in Technology Development - The Case of Micro Process Engineering

by

Dipl.-Ing. Eva Zschieschang
geboren in Dresden

Vom Fachbereich Bauingenieurwesen und Geodäsie der
Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines Doktor-Ingenieurs
(Dr.-Ing.)
genehmigte Dissertation.

Tag der Einreichung: 18.01.2013
Tag der mündlichen Prüfung: 15.03.2013

D17
Darmstadt, 2013

Referenten:

Prof. Dr. rer. nat. Liselotte Schebek, TU Darmstadt
Prof. Dr.-Ing. Thomas Turek, TU Clausthal

Erfolg ist, seiner Zeit voraus zu sein und trotzdem schon verstanden zu werden.

Bernd Lötsch

Erklärung zur Einhaltung der Grundsätze wissenschaftlichen Arbeitens

Hiermit versichere ich, die vorliegende Arbeit selbstständig angefertigt, sämtliche Zitate gekennzeichnet und Literaturquellen sowie verwendete Hilfsmittel vollständig angegeben zu haben.

Für meine Oma Ursula Silbermann

ACKNOWLEDGEMENTS

I am grateful for the support of my advisors Liselotte Schebek (TU Darmstadt) and Peter Pfeifer (Institute for Micro Process Engineering, KIT), who made this work possible.

This thesis was performed at the Institute for Technology Assessment and Systems Analysis (ITAS), Karlsruhe Institute for Technology (KIT). I want to thank all members of the institute for hospitality and support.

Thanks to all members of the *Simulanten* group at ITAS who embarked in the adventure to establish a venue for modeling and simulation in systems analysis. It was a pleasure to work with such motivated people creating an incredibly inspiring environment. In particular, I would like to thank Marcel Weil, Jens Buchgeister, Udo Jeske, Bernardo Cienfuegos, Oliver Hurtig and my two students Simon Fromme and Manuel Selinsek.

Special thanks go to my family who enabled my scientific career by giving me the freedom of choice and the support for realization.

Finally, I am deeply grateful for the loving support of Clemens Grabher who was always there for me and never lost faith in my ideas. Thank you so much, Clemens!

TABLE OF CONTENTS

DEDICATION v

ACKNOWLEDGEMENTS vi

LIST OF FIGURES xi

LIST OF TABLES xv

LIST OF APPENDICES xvii

LIST OF ABBREVIATIONS xix

GLOSSARY xxi

ABSTRACT xxv

ZUSAMMENFASSUNG xxvii

CHAPTERS

1. Introduction 1

1.1 Background 1
1.2 Problem definition 2
1.3 Objectives 3
1.4 Thesis framework 3

2. Background 5

2.1 Life cycle assessment 5
2.1.1 Methodology 6
2.1.2 Software in LCA 9
2.1.3 Methodological issues beyond state-of-the-art LCA . . . 10
2.1.4 LCA method guideline 11
2.1.5 LCA in technology development 11
2.1.6 Key issues for using LCA 13

2.2 Micro process engineering . . . 15
2.2.1 Micro reaction engineering . . . 15
2.2.2 Paradigm shift and novel process windows . . . 20
2.2.3 Technology development . . . 21
2.3 Case study . . . 25
2.3.1 Introduction . . . 25
2.3.2 Chemical concepts . . . 26
2.3.3 Catalysts development . . . 27
2.3.4 Synthesis gas production . . . 28
2.3.5 Gas-to-Liquid process . . . 28
2.3.6 Application fields and potentials . . . 29
2.4 Challenges for LCA in microreactor development . . . 31

3. Systems analysis of complex systems . . . 33

3.1 General systems theory . . . 33
3.1.1 Basics . . . 33
3.1.2 System description . . . 34
3.1.3 Modeling . . . 36
3.2 Sensitivity model . . . 37
3.3 Object-oriented software engineering . . . 38

4. New methodology for applying LCA in technology development . . . 41

4.1 Simplified microreactor development model . . . 41
4.1.1 Modeling framework . . . 41
4.1.2 Interdependencies in the general systems theory . . . 43
4.2 Requirements towards the new methodology . . . 45
4.3 Modular Server-Cient-Server methodology . . . 47
4.3.1 Modularization . . . 48
4.3.2 Server and client . . . 49
4.3.3 Interfaces . . . 51
4.4 Workflow for combined application . . . 53
4.5 LCA software development . . . 55

5. Case Study . . . 57

5.1 Research questions in microreactor development . . . 57
5.2 Detailed microreactor development model . . . 59
5.3 Microreactor . . . 63
5.3.1 Introduction . . . 63
5.3.2 Methodology . . . 63
5.3.3 Results . . . 73
5.3.4 Discussion . . . 78

5.4 Fischer-Tropsch synthesis . . . 79
5.4.1 Introduction . . . 79
5.4.2 Methodology . . . 79
5.4.3 Results . . . 82
5.4.4 Discussion . . . 85
5.5 Gas-to-Liquid process . . . 87
5.5.1 Introduction . . . 87
5.5.2 Methodology . . . 87
5.5.3 Results . . . 90
5.5.4 Discussion . . . 98
5.6 Offshore application of the GtL process . . . 99
5.6.1 Introduction . . . 99
5.6.2 Methodology . . . 100
5.6.3 Results . . . 104
5.6.4 Discussion . . . 107
5.7 Systems analysis . . . 109
5.7.1 Microreactor development . . . 109
5.7.2 Manufacturing and use phase of a microreactor . . . 111
5.7.3 Application of microreactors to replace gas flaring . . . 112
5.7.4 Discussion . . . 114

6. Discussion . . . 115

7. Outlook & Conclusion . . . 119

7.1 Outlook . . . 119
7.2 Conclusion . . . 120

APPENDICES . . . 121

BIBLIOGRAPHY . . . 173

LIST OF FIGURES

Figure

2.1 LCA phases. 6

2.2 Flow chart of a simplified product system with unit processes. 8

2.3 LCA methodologies related to the main phases of the LCA. 12

2.4 Interdependencies in microreactor development. 16

2.5 Microreactor components. 17

2.6 Microreactor interior in plate design and the relevant parameters. 18

2.7 Interdependencies in the microreactor development process from predesign to manufacturing. 20

2.8 Course of three important technology development parameters during individual technology development phases. 21

2.9 Time-to-market of processes and products for established approaches and approaches including microreactors. 22

2.10 Dependencies between the technology development phase, decision maker, interests and the life cycle of a microreactor. 23

2.11 Dependencies of specific parameters, boundary conditions, design and process requirements, and the decision making. 24

2.12 GtL process chain with main components. 28

3.1 General procedure of the system analysis. 33

3.2 Schematic representation of the formal system definition. 35

3.3 System concepts in practice. 35

4.1 Simplified microreactor development model. 42

4.2 Interconnection within the same system hierarchy. 43

4.3 Interconnection between different system hierarchies. 44

4.4 Differences in modeling a given system in LCA and R&D. 45

4.5 Interdependencies between R&D and the life cycle of a microreactor. . . . 46

4.6 Vertical and horizontal modularization of a system. 48

4.7 Module structure with different attributes and methods. 49

4.8 Content and software level within the Modular Server-Client-Server (MSCS) methodology. 50

4.9 Module set according to the MSCS methodology. 54

5.1 Detailed microreactor development model based on the general systems theory definitions. 60

5.2 Simplified flow-chart of the LCA model for the microreactor fabrication. . 69

5.3 R&D, interface and LCI parameters for wet chemical etching and milling. 71

5.4 Modular Server-Client-Server methodology applied to the example of microreactor manufacturing. 72

5.5 Total GWP impact of three different fabrication processes for one design of a microreactor. 73

5.6 Impact of the individual fabrication steps using wet chemical etching as structuring process. 74

5.7 GWP Impacts of different sources during microreactor fabrication. 75

5.8 Impact of the design parameter catalyst void fraction on the total GWP in microreactor fabrication. 76

5.9 Impact of design and process parameters using chemical etching as structuring process. 76

5.10 Impact of design and process parameters in the milling process. 76

5.11 GWP for microreactors with different production rates. 77

5.12 Amount of steel used for the microreactor components. 77

5.13 LCA model of the FTS process. 81

5.14 Calculated chain growth probability. 82

5.15 Chain growth probability as function of temperature and pressure. 83

5.16 GWP of the FTS process. 85

5.17 Simplified process flow diagram of a GtL process. 87

5.18 LCA black box model of the GtL process. 90

5.19 Reactant material flow. 91

5.20 Product material flow. 92

5.21 Reactant and product composition. 92

5.22 Normalised GWP results for scenario 1 - 24 sorted according to chain growth probability variations for the GtL process. 95

5.23 Normalized GWP results for scenario 1 - 24 sorted according to CR variations for the GtL process. 96

5.24 Normalized GWP results for scenario 1 - 24 sorted according to pressure variations for the GtL process. 97

5.25 Normalized GWP results for three reactor sizes for scenario 1 - 24 of the GtL process. 97

5.26 System boundaries for gas flaring and offshore GtL. 99

5.27 Influence of the oil-price variation. 105

5.28 Variation in CO_2 certificate price. 105

5.29 GtL plant cost variation. 106

5.30 GtL plant scaling for two GtL process configurations. 106

5.31 Relations between hierarchical systems of the microreactor development. 109

5.32 GWP change with reactor life time (chemical etching as structuring process). 112

5.33 GWP change with reactor life time (milling as structuring process). 113

5.34 GWP for three GtL process scales in offshore application converting 9 million kg methane per year compared to gas flaring and venting. These GtL processes are described by the designs and the manufacturing technique of the considered microreactor. 114

B.1 Data flows between R&D modules in microreactor development. 129

B.2 Data flows within manufacturing module. 130

B.3 Network of all manufacturing processes using chemical etching for reaction and cooling channel structuring in Umberto®. 131

B.4 Network of all manufacturing processes using milling for reaction and cooling channel structuring in Umberto®. 132

B.5 Network of all manufacturing processes using chemical etching for reaction and milling for cooling channel structuring in Umberto®. 133

B.6 Network of the assembly processes. 134

D.1 Aspen model for the GtL process. 154

LIST OF TABLES

Table

2.1 Key issues for LCA application. 13

2.2 Manufacturing techniques for microreactors in plate design. 19

5.1 Questions related to LCA application within the three microreactor development phases and the application field. 58

5.2 Modeling parameters for microreactor design, structuring, and assembly and bonding. 68

5.3 Variation of design parameters for FTS 81

5.4 Reactant and products for different selectivities and chain growth probabilities. 84

5.5 Associated lean gas composition 89

5.6 Reactants of the simulated GtL process. 91

5.7 Product of the simulated GtL process. 93

5.8 Reactant and Product data of scenario 1 for the simulated GtL process. . . 94

5.9 Cost and GWP optimization parameters. 100

5.10 GWP values for manufacturing one FTS microreactor design and for two GtL processes. 103

5.11 GtL plant prices for three plant scales. 103

B.1 Microreactor design parameters for the manufacturing processes. 128

C.1 Calculated product distribution and weight of the FTS products. 152

D.1 Scenario parameters for Aspen modeling. 157

LIST OF APPENDICES

Appendix

A. Formal systems definition 123

B. Manufacturing 127

C. Fischer-Tropsch synthesis 149

D. Gas-to-Liquid 153

E. Optimization 159

LIST OF ABBREVIATIONS

ASF	Anderson-Schulz-Flory
ATR	Autothermal Reforming
BCM	billion cubic meters
CAD	Computer Aided Design
CALCAS	Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability
CFD	Computational Fluid Dynamics
COBRA	Common Object Request Broker Architecture
CR	Conversion Rate
DCOM	Distributed Component Object Model
EnviroInfo	International Conference on Informatics for Environmental Protection
FTS	Fischer-Tropsch synthesis
GGFR	Global Gas Flaring Reduction
GtL	Gas-to-Liquid
HTFT	High Temperature Fischer-Tropsch synthesis
HTTP	Hypertext Transfer Protocol
IMVT	Institute for Micro Process Engineering
IO	Input-Output
ISIE	International Society for Industrial Ecology
ISO	International Organization for Standardization
ITAS	Institute for Technology Assessment and Systems Analysis
KIT	Karlsruhe Institute for Technology
LCA	Life Cycle Assessment
LCE	Life Cycle Engineering
LCI	Life Cycle Inventory
LCIA	Life Cycle Impact Assessment
LPG	Liquefied Petroleum Gas
LTFT	Low Temperature Fischer-Tropsch synthesis
MSCS	Modular Server-Client-Server

NGL	Natural Gas Liquids
NPW	Novel Process Window
R&D	Research and Development
SETAC	Society of Environmental Toxicology and Chemistry
syngas	synthesis gas
UNCED	United Nations Conference on Environment and Development
UPLCI	Unit Process Life Cycle Inventory
WIG	Wolfram Inert Gas welding
WGS	Water-Gas-Shift
XML	Extensible Markup Language

GLOSSARY

background process — A process necessary for the flow chart analysis in LCI but with no special interest is called background process. Generic data are sufficient for background processes.

black box — Black box in the context of LCI modeling is understood as process described by the in- and output flows without any further detailed knowledge about functional relations of the interior.

device — A device is a part of the technology under development.

element — An element is a part of a system.

endpoint impact assessment — Endpoint impact assessment uses impact indicators of an impact category from the end of the cause-effect chain of an environmental mechanism.

foreground process — A process analyzed in detail within a flow chart in LCI is called foreground process. Foreground processes require process-specific data.

functional unit — The functional unit is the key element of the LCA, in which the function of the studied product system is defined. The functional unit provides a reference to which the in- and outputs of the product system can be related and therefore allows the comparison of different product systems with the same functional unit.

general systems theory	Theory about the nature of a system which was originally proposed by the biologist Ludwig von Bertalanffy. According to the general systems theory a system is characterized by the interactions of its elements.
microreactor	A microreactor is a reactor for substance conversion characterized by an interior structure in the micrometer scale and therefore with a high surface-to-volume ratio.
methodology	A methodology is a guideline containing a set of principles and rules on how to solve a discipline specific problem.
midpoint impact assessment	Midpoint impact assessment uses impact indicators of an impact category prior to the endpoints in the cause-effect chain of an environmental mechanism.
model	Models are earmarked transformations of systems.
modularization	The decomposition of a system into smaller parts is called modularization and the small parts are called modules.
product	In LCA, a product is defined as any good or service.
stakeholder	A stakeholder is a person or group having strong interest in the course or outcome of a process or project. This could be the society, investors in general, a company, the head of R&D or responsible engineers for the example of technology development.

system	A system is described as organization of elements and their relation to each other within defined system boundary. In accordance to the in- and outputs between the system and the surrounding area, two system types are distinguished: the open and closed systems. The open system has in- and outputs to the surrounding area while the closed system has none.
system complexity	The system complexity is described as the quotient between the number of relations (n_R) and the number of elements (n_E) of the considered system.
technology development	Technology development describes the process from the initial idea to the final technology.
unit process	A unit process is the smallest process unit of the flow chart, without any further processes included. In the context of LCA, a unit process is defined as black box process.

ABSTRACT

Life Cycle Assessment in Technology Development
- The Case of Micro Process Engineering
by
Eva Zschieschang

Chair: Prof. Dr. Liselotte Schebek

Technology development is a sophisticated process including multiple development phases. Already the research phase as the first phase of the development process can significantly determine the later technical, economical and environmental performance of a technology. Due to the high environmental impact of technologies during their life cycle, the use of sustainability analysis methodologies in an early stage of technology development seems valuable.

Life Cycle Assessment (LCA) is a methodology to analyze the environmental impact of products during their whole life cycle. This methodology was originally developed to compare different yet functionally analogous products, such as tetra packs and glass bottles. In the last 20 years, life cycle based product analysis became increasingly popular and was adapted to new application fields requiring adaptation also of the LCA methodology itself. Thus, multiple new approaches and methodologies were developed. Due to the lack of a uniform scientific LCA approach, a general LCA methodology towards the analysis of different product types and application fields (i.e. technology development) is currently unavailable.

The present work examines the application of LCA in technology development. The study includes an analysis of existing methodological approaches, and the identification of technology-specific requirements for the LCA on the example of a case study. In addition, a new method for the application of LCA in the development of a new technology was designed and successfully applied to the case study. As an example for the development of a complex technology, a microreactor for Fischer-Tropsch synthesis (FTS) in a Gas-to-Liquid

(GtL) process was chosen as a case study. The focus of the theoretical work lies in the modeling of the technology development process and how to deal with different models in LCA and technology development. Based on the general systems theory, a model of micro-reactor development was created. Subsequently, a method for the application of LCA on the model of micro-reactor development was developed. Using this novel methodology, four phases of the microreactor development were analyzed by LCA; design and manufacturing, use of a microreactor for the FTS, application of the microreactor in the GtL process and application of the GtL process for the substitution of gas flaring in offshore oil production.

The LCA results obtained with the new methodology provide evidence on environmental effects of various design configurations of the technology under development throughout the development process. The application of this new methodology during the microreactor development process for the first time allows the creation of environmentally optimized GtL processes for site-specific oil production.

ZUSAMMENFASSUNG

Anwendung der Ökobilanzierung in der Technologieentwicklung
am Beispiel der Mikroverfahrenstechnik

von

Eva Zschieschang

Erstgutachter: Prof. Dr. Liselotte Schebek

Die Neu- und Weiterentwicklung von Technologien ist ein sehr langwieriger und komplexer Prozess, der sich in verschiedene Entwicklungsphasen untergliedert. Schon in der ersten Phase dieses Entwicklungsprozesses werden die Weichen für die spätere technische, ökonomische sowie die ökologische Performance der Technologie gestellt. Da die ökologischen Auswirkungen von Technologien einen großen Beitrag an den heutigen Umweltproblemen haben, ist die Anwendung von Analysemethoden zur Nachhaltigkeitsuntersuchung von Technologien in der frühen Entwicklungsphase unabdingbar.

Die Ökobilanzierung ist eine Analysemethode zur Nachhaltigkeitsuntersuchung. Mit dieser Methode werden die ökologischen Auswirkungen während des gesamten Lebensweges eines Produktes analysiert. Die Ökobilanzierung wurde ursprünglich entwickelt, um unterschiedliche Produkte mit gleicher Funktion nach ökologischen Gesichtspunkten zu vergleichen, z.B. Tetrapacks und Glasflaschen. Im Laufe der letzten 20 Jahre wurde diese lebenszyklus-basierte Analysemethode immer populärer. Eine Erweiterung des Anwendungsspektrums machte jedoch eine Weiterentwicklung der Methode erforderlich. Es entstanden daher viele neue methodische Ansätze. Weil jedoch der Ökobilanzierung bis heute keine einheitliche wissenschaftliche Basis zu Grunde liegt, ist die Anwendung dieser Methode für neue Produkttypen und Anwendungsfelder, wie zum Beispiel die Technologieentwicklung, keineswegs trivial.

In der vorliegenden Arbeit wird die Anwendung der Ökobilanzierung in der Technologieentwicklung untersucht. Die Untersuchung beinhaltet die Analyse bestehender methodischer Ansätze, sowie die Identifizierung von technologiespezifischen Anforderungen

an die Ökobilanzierung anhand eines Fallbeispiels. Im Weiteren wird eine neue Methode zur Anwendung der Ökobilanzierung in der Technologieentwicklung entworfen, und diese Methode erfolgreich auf das Fallbeispiel angewendet. Als Fallbeispiel für eine komplexe Technologie wird die Entwicklung eines Mikroreaktors für die Fischer-Tropsch Synthese in einem Gas-to-Liquid Prozess ausgewählt. Der theoretische Schwerpunkt der Arbeit liegt in der Modellierung des Technologieentwicklungsprozesses und dem Umgang mit unterschiedlichen Modellen in Ökobilanzierung und Technologieentwicklung. Auf Basis der allgemeinen Systemtheorie wird ein Modell der Mikroreaktorentwicklung erstellt. Mit Hilfe der neu entwickelten Methode zur Anwendung des Technologieentwicklungsmodells in der Ökobilanzierung werden vier Entwicklungsphasen der Mikroreaktorentwicklung untersucht; die Konstruktion und Fertigung, die Anwendung eines Mikroreaktors für die Fischer-Tropsch Synthese, die Anwendung des Mikroreaktors im GtL-Prozess und die Anwendung des GtL-Prozesses zur Substitution der Erdgasverbrennung bei der Ölförderung auf Bohrinseln.

Die gewonnenen Ökobilanzergebnisse liefern Erkenntnisse zu Umweltwirkungen verschiedener Designkonfigurationen der zu entwickelnden Technologie über den gesamten Entwicklungsprozess. Die Anwendung dieser neu entwickelten Methode in der Mikroreaktorentwicklung erlaubt nun die Entwicklung von ökologisch optimierten GtL-Prozessen für standortspezifische Ölförderung.

CHAPTER 1

Introduction

1.1 Background

Growing awareness of man-made environmental changes during the seventies and eighties of the 20th century led to the first United Nations Conference on Environment and Development (UNCED) in Rio de Janeiro, Brazil initiating a global discussion on sustainable development (*UNCED*, 1999). This also marked the starting point for the implementation of the concept of sustainable development in various public, economic and technical areas of application through legal regulations.

Especially in technological areas sustainable development is of major relevance due to the high environmental impact of technologies during their life cycle.

Technology development is a highly sophisticated and expensive process characterized by four properties specific to the technology to be developed which are: 1) the degree of freedom, 2) state of the art, 3) possible system integration, and 4) specific requirements of decision makers towards the technology of interest (*Peters et al.*, 2003, *Eversheim*, 2005). The whole development process consists of several phases and can take up to years (*Kockmann*, 2006). Already the research phase as the first phase of the development process can significantly determine the later technical and environmental performance of a technology. Therefore, legislative authorities have promulgated regulations towards sustainable development particularly affecting this early phase of development (*Commission of the European Communities*, 2001, *European Parliament and Council*, 2005).

Micro reaction engineering, an emerging technology and part of micro process engineering, promises economic and environmental advantages through Novel Process Window (NPW), which are enabled by micro reaction engineering-specific characteristics, resulting in reduced costs (i.e. through shortened development process), reduced resource demands and waste production through higher efficiency and selectivity (*Hessel*, 2009, *Mi-*

tropetros and Bazzanella, 2010, *Hessel*, 2011). The use of sustainability analysis methodologies in an early-stage technology development seems valuable to verify these promising advantages, especially in application fields where micro reaction engineering is supposed to replace conventional processes with the aim to reduce environmental impacts. The use of micro reaction engineering to replace gas flaring could be one such application field.

1.2 Problem definition

Integrated Process Development (*Hungerbühler*, 1999), and Life Cycle and Eco Design (*Abele*, 2008) are accepted sustainable concepts in technology development using the methodology of Life Cycle Assessment (LCA) (*DIN EN ISO 14040*, 2006, *DIN EN ISO 14044*, 2006). This methodology promises high environmental and economic potential in early-stage technology development, through identification of so called environmental hot spots (i.e. energy consuming processes) (*Hessel*, 2009). As a consequence, a more sustainable developmental path can be implemented with lower total costs compared to changes in later development phases.

First methodological research towards LCA in early-stage technology development was initiated with the beginning of the 21st century. Major challenges for this new approach include system complexity, high data uncertainty or lack of data, especially within the manufacturing, use and recycling phase for new technologies. Currently, two approaches for dealing with data uncertainty and lack of data are applied. The first approach uses existing data for a simplified/reduced LCA (*Rydh*, 2005, *Guinée*, 2002, *Hochschorner*, 2003, *Park et al.*, 2006, *Hübschmann et al.*, 2009), whereas the second approach tries to calculate the missing data (*Müller et al.*, 2004, *Diaz*, 2010, *Yuan and Dornfeld*, 2010, *Schrems et al.*, 2011).

The system complexity, generally characterized by the number of elements and their interdependencies (*Ropohl*, 1979), is of major relevance for micro process engineering due to the high interdependency between device and process development. Screening approaches are one tool for reducing the system complexity by reducing the number of elements for decision making in early-stage technology development (*Gasafi*, 2006).

Importantly, micro reaction engineering, as compared to conventional engineering, introduces a paradigm shift from process-oriented to reaction-oriented device and process development (*Kockmann*, 2006). This paradigm shift requires a hand-in-hand design of the microreactors (devices) and processes, which leads to a high interdependency between device and process development (*Hessel*, 2009).

The application of Life Cycle Assessment for interdependent device and process deve-

lopment in general and in micro reaction engineering in particular has not thoroughly been discussed yet. So far, only few case studies exists for LCA in micro reaction engineering (*Kralisch and Kreisel*, 2007, *Hessel et al.*, 2008, *Hübschmann et al.*, 2009, 2011). These case studies utilize existing microreactors. One study compares continuous (microreactor) with batch (conventional double-walled reactor) processes (*Kralisch and Kreisel*, 2007) and the other study optimizes the process design of one microreactor in terms of reaction conditions such as temperature, pressure, solvents and heating (*Hübschmann et al.*, 2009). Neither the development of the microreactor nor the consequences for the process are considered in these studies.

1.3 Objectives

The aim of this thesis is to develop an appropriate methodology for the analysis of interdependent device and process development from a life cycle perspective by applying state of the art concepts for interdependent systems to LCA, and validate this on the example of micro reaction engineering.

In particular, it is examined how current life cycle assessment methodologies may be applied within interdependent device and process development in micro process engineering aiming at a replacement of gas flaring in oil production. The process under investigation is Gas-to-Liquid fuel conversion via Fischer-Tropsch synthesis through micro reaction engineering. The focus lays on the early research phase where the degree of freedom and therefore the impact of decision-making is higher than at later stages.

The developed methodology shall allow early evaluation of different microreactor designs within different process designs for offshore application to replace gas flaring from a life cycle perspective. Thereby, it contributes to a more sustainable product and process development due to reduced total development time, costs and resources. Furthermore, the methodology shall utilize common research software-tools and therefore supports an easy integration in device and process design which may result in higher acceptance and application in industry.

1.4 Thesis framework

This thesis is organized in seven chapters.

The second chapter provides background information on the topics at the core of the research. It introduces the concept of life cycle assessment with focus on the application in

technology development, micro process engineering as well as the case study of the Gas-to-Liquid process in microreactors for interdependent device and process development.

The third chapter provides an overview on state of the art approaches concerning interdependent systems.

The fourth chapter examines requirements for the use of life cycle assessment in the case study. Furthermore, it presents the modeling framework for technology development and the methodology for the application of life cycle assessment in interdependent device and process development based on the results of the previous chapter.

The case study is examined with the novel methodology in the fifth chapter.

The final two chapters provide a summary of the research comprising this thesis, its main conclusions and suggests further research to improve sustainable technology development.

CHAPTER 2

Background

This chapter introduces the main subject of this interdisciplinary work. Section 2.1 provides an overview of the methodology of Life Cycle Assessment with focus on the application in technology development. The technology to be analyzed with the LCA methodology is described in Section 2.2 and the considered case study of the Fischer-Tropsch synthesis as part of the Gas-to-Liquid process in Section 2.3. Finally Section 2.4 summarizes the challenges for LCA in technology development and concretize the research question of this thesis.

2.1 Life cycle assessment

The standardized methodology of LCA addresses environmental aspects and potential environmental impacts throughout a products life cycle from raw material acquisition through production, use, end-of life treatment, recycling and final disposal. Such analyses contribute to a better understanding of systemic interdependencies and enable to address these impacts (*DIN EN ISO 14040*, 2006, *DIN EN ISO 14044*, 2006). A product is here defined as any good or service.

LCA is usually applied for:

1. Identification of hot spots within the products life cycle
2. Comparison of different products with the same function
3. Comparison of different design, manufacturing, process, and material options within one products life cycle with the aim of lowering the environmental impact

2.1.1 Methodology

The Life Cycle Assessment methodology consists of four phases: goal and scope definition, Life Cycle Inventory (LCI) analysis, Life Cycle Impact Assessment (LCIA), and interpretation. The first phase determines the depth and width of the LCA by setting the system boundary and level of detail. The incoming and outgoing material and energy flows of the product system under investigation are collected and modeled in the second phase (LCI). These results are assigned to the selected impact category representing environmental issues of concern in the third phase (LCIA). Finally, the inventory and impact assessment results are summarized and discussed with respect to the goal and scope definition of the first phase. LCA is an iterative process because all phases are repeated several times. Figure 2.1 illustrates the iterative approach of the LCA methodology with the dependencies between the four phases.

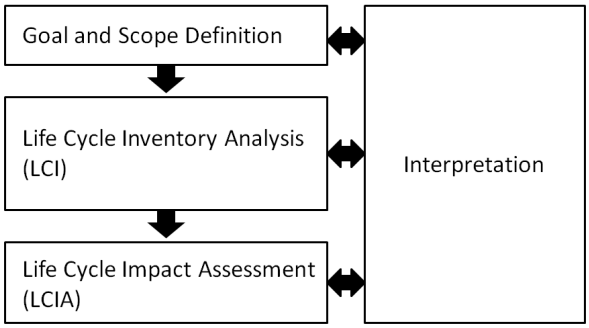

Figure 2.1: LCA phases according to (*DIN EN ISO 14040*, 2006).

2.1.1.1 Goal and scope definition

Within the first step of the goal and scope definition, the goal of the LCA study would ideally be defined by both, the commissioner and the practitioner together. The goal would include the intended application and audience, the reasons for carrying out the study, and whether the results are intended to be used for comparative assertions (*DIN EN ISO 14040*, 2006). Based on that, the scope of the study is determined in the second step, including defining the functional unit, choice of impact categories and method of impact assessment, system boundaries, type of LCA (attributional, consequential), data quality requirements, and system subdivision in fore- and background processes (*Baumann*, 2009). All decisions made in this step are related to the modeling of the considered product system to be carried out in LCI.

2.1.1.2 LCI

Within LCA, the LCI phase requires the highest effort and time (*Joint Research Center*, 2010, *DIN EN ISO 14040*, 2006). This phase consists of four parts subdivided into several work packages. The four parts are: construction of the flow chart of the considered product system , data collection, data selection and averaging, and finally system calculation of LCI results (*Joint Research Center*, 2010).

Modeling the product system by construction of the flow charts of the considered processes is the first phase within LCI. The flow chart of the product system consists of several processes; the smallest element is called unit process. A unit process is a black box process, defined by its in- and output flows without any knowledge of the internal working, as illustrated in Figure 2.2. LCI flow chart models are usually static and linear, which means no time factor is included and the polynomial function between in- and output flows has zeroth or first degree (*Baumann*, 2009). Linear polynomial functions are continuous and differentiable.

Attributional and consequential modeling are the two main modeling principles in LCI practice. The main difference results from the technosphere of the considered system, which is static for attributional and dynamic for consequential modeling. Consequential modeling *'aims at identifying the consequences that a decision in the foreground system has for other processes and systems of the economy, both in the analyzed system's background system and on other systems'* (*Joint Research Center*, 2010, p.71-72). The ILCD handbook describes in detail how consequential processes can be identified, but it lacks specific mathematical modeling and calculation methods for this.

In addition to attributional and consequential LCI, a third type is the so called parameterized LCI (*Müller et al.*, 2004). Here, the inventory of the whole system or single subsystems is parameterized by specific characteristics. Within LCI these characteristics, called parameters, lead to the following advantages compared to conventional inventory: higher flexibility for database use (parameterized database), sophisticated scenario modeling, and interfaces for mathematical simulations or sensitivity analysis (*Schmidt*, 2007). Therefore, parameterized LCI shows a high potential as analysis tool in technology development, especially in the early design phase (*Müller et al.*, 2004).

Data collection, the second part within LCI, is the most important, since modeling results are based on these data. During data collection, quantitative and qualitative data are collected for the considered product system. In general, LCA models consists of foreground and background system, where the foreground system contains one or more processes to be analyzed in detail. The processes are called foreground respectively background pro-

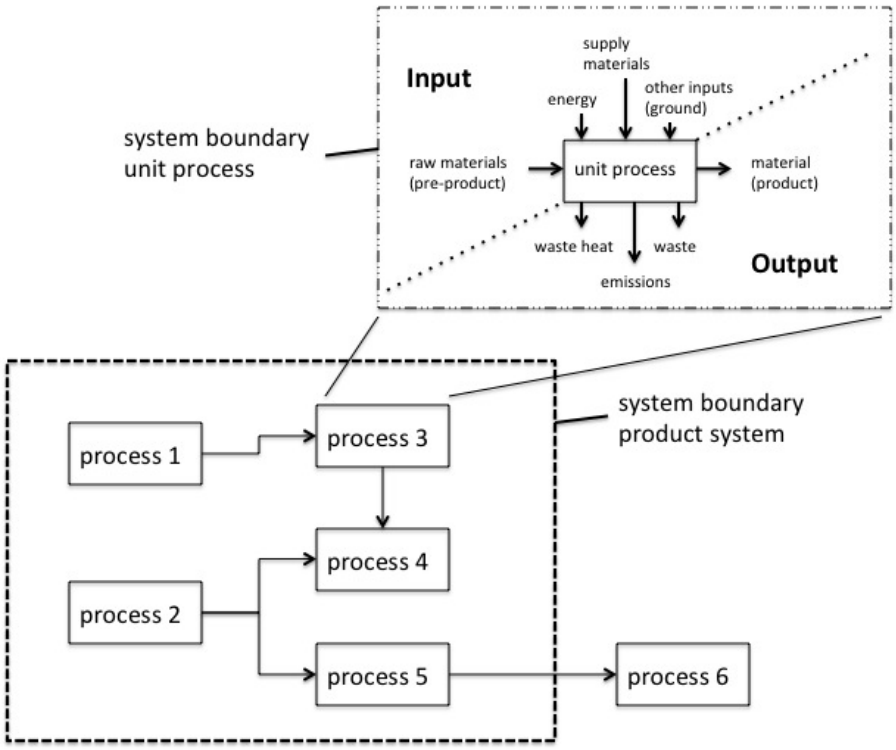

Figure 2.2: Flow chart of a simplified product system with unit processes (*DIN EN ISO 14040*, 2006, *Klöpffer*, 2009).

cesses. Data type and source differ within these two process types. Process-specific data are usually used for foreground processes whereas generic data are sufficient for background processes. These data are collected through measuring existing material and energy flows of the product system. The data quality, comprising accuracy (i.e. representativeness), precision (i.e. uncertainty) and completeness of the inventory, determines the overall quality of the LCA and depends on the goal and scope and the level of detail required.

Data selection and averaging are realized in the third phase of LCI according to the goal and scope defined in the first phase of the LCA.

Upon completion of data collection and selection, the final flow chart calculation can be conducted in the fourth phase. The entire calculation procedure consists of three steps; normalizing data according to an amount of product, calculation of the product flows with respect to the functional unit as reference flow, and summary and documentation of the calculated data. System calculation of the LCI results follows in- and output material flow analysis of (black box) unit-processes. Black box in the context of LCI modeling is understood as unit process containing multi-functional processes, in contrast to mono-functional unit processes (*Joint Research Center*, 2010). Basic scientific laws considered

in LCI calculation are: preservation of mass, first and second principal of thermodynamics, stoichiometry, and equivalence of mass and energy (*Klöpffer*, 2009).

2.1.1.3 LCIA

In LCIA, inventory results are associated with specific environmental impacts following the cause-effect chain. Inventory data are classified into impact categories and category indicators and then associated with environmental impacts by using characterization methods and factors. A common and widely used characterization method is the global warming caused by green house gases with the gas specific characterization factor *Global Warming Potential (GWP)* quantified in kg CO_2-Eq.per kg gas. Impact categories are either midpoint or endpoint categories. In practice, for each impact category and characterization model a variety of LCIA methods exist: for example CML2001 as midpoint (*Guinee*, 2001a,b), Eco-Indicator 99 as endpoint (*Goedkoop and Spriensma*, 2000, 2001), and Impact 2002 for both impact categories (*Jolliet et al.*, 2003).

2.1.1.4 Interpretation

The interpretation phase of the LCA analysis brings together results from LCI and LCIA with respect to the goal and scope defined. Within the interpretation phase, two analysis types can be distinguished; first, analyzing the results with focus on significant issues (most energy consuming process, influence of methodological choices) and second, analyzing the results with focus on the robustness of the results.

2.1.2 Software in LCA

The LCA methodology provides a guideline on how to do a life cycle analysis of a considered product system. For the practical implementation various software tools are necessary. In principle there are two types of software tools; first, databases for collecting inventory data in LCI as well as characterization and weighted factors in LCIA and second, software tools for calculating the inventory of the product system in LCI and the environmental impact in LCIA. A variety of software packages, each tailored to a particular application field, is currently available on the market. Umberto® (*Hedemann*, 1998-2012), GaBi (*Betz*, 2012), Sima-Pro (*Goedkoop*, 2012) and OpenLCA (*Ciroth et al.*, 2012) are examples for common LCA software packages for a full LCA including LCI and LCIA. The major database for LCA used in Europe is the commercial 'Ecoinvent' database, which provides generic data for LCI in the EcoSpold format (*Weidema and Hischier*, 2010). This common data exchange format is based on Extensible Markup Language (XML) and is

fully ISO/TS 14048 compliant. The database contains several datasets on a unit process level linking in- and output flows to the functional unit (unit process: steel production, functional unit: 1 kg steel).

2.1.3 Methodological issues beyond state-of-the-art LCA

Due to increasing number of application fields adapting LCA, the methodology itself is under permanent development. Besides LCA performed according to ISO norms, various other approaches have been developed. These approaches take into account the economic and social aspects of sustainability (Life Cycle Costing, Social Life Cycle Assessment), and integrate mechanism such as rebound and price effects (consequential LCA, attributional LCA), as well as time (steady-state LCA, dynamic LCA) and space dimensions (spatially independent -global- and differentiated - regional-) (*Guinee et al.*, 2006). Including such approaches into LCA methodology is known as *deepening and broadening LCA* (*Schepelmann et al.*, 2006). These new methodologies have to face the challenge of satisfying the desire for simplified tools and methods despite the simultaneous increase in system complexity and amount of data.

Currently, four major conferences focusing on LCA methodology and application exist; *Society of Environmental Toxicology and Chemistry (SETAC)*, *Life Cycle Engineering (LCE)*, *International Society for Industrial Ecology (ISIE)* and *International Conference on Informatics for Environmental Protection (EnviroInfo)*. Major scientific journals in this field include *International Journal of LCA*, *Journal of Cleaner Production* and *Industrial Ecology*. A look at the scope of these conferences and journals illustrates the diversity in LCA methodology. Whereas LCE and ISIE focus on engineering and industrial application fields, EnviroInfo is devoted to software developments and SETAC covers the whole methodology as stated in the International Organization for Standardization (ISO) norms (*DIN EN ISO 14040*, 2006, *DIN EN ISO 14044*, 2006).

In order to coordinate these developments, the European Union initialized the 6th Framework Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability (CALCAS), with the aim to categorize the LCA approaches that have emerged (*CALCAS*, 2006). Results of this project show the need for a scientific framework besides the current procedural framework of ISO standards for LCA: This procedural framework '*served a definite function in facilitating the communication between scientists, practitioners, and others by providing a vocabulary, and in pointing out points of agreement and disagreement. But they have also a limited meaning or even failed in aspects such as not providing a scientific basis, not providing the intended clarity that is needed for a routine application,*

and not providing the indispensable data and formulas' (*Heijungs et al.*, 2006, p.18) .

In addition to the introduction of a new scientific framework, '*further improvements relate to better guidance on how to define questions and link them to most adequate levels of modeling, of which the ISO type is only one, the most easily applicable. Also, the interpretation stage can be improved by a more systematic uncertainty analysis, covering both data uncertainty and model uncertainty, and by further developing quantified methods for sensitivity analysis, contribution analysis; and by linking to multi-criteria analysis, including formalized weighting procedures*' (*Guinee et al.*, 2006, p.9).

Today, the main modeling types in LCI are steady state and comparative static equilibrium analysis, however, both of which do not include the factor of time. Similarly, '*neither the causal models nor the scenario based models are available for the operational analyses and evaluation of technologies*' (*Guinee et al.*, 2006, p.9). The level of detail requirements within this application field exceeds the capacity of the current form of in- and output material flow analysis in LCI.

In conclusion, current ISO-standard LCA is subject to numerous further developments on both scientific and application levels.

2.1.4 LCA method guideline

In recent years, many authors and users of LCA tried to go beyond the ISO standards, and therefore multiple methodologies have been developed. Reviewing these methodologies in terms of general use (standard vs. simplified), inventory methodologies (attributional, consequential, Input-Output (IO), Hybrid Analysis) and impact assessment methods allowed to classify the major existing methodologies for different LCA phases as shown in Figure 2.3. Moreover, Figure 2.3 provides a guideline for the use of LCA in new application fields. For example: each methodology selected in the goal and scope phase must contain one or more methodologies from the second and third phase, i.e. a full LCA could use a consequential LCI and a midpoint impact assessment method. Consequently, no general LCA methodology for a specific application field can be proposed. Instead, each LCA phase requires a specific method, taking into account the specific requirements of the application field.

2.1.5 LCA in technology development

Literature on LCA in technology development either focuses on one single technology development phase (*Lindner*, 2010), on manufacturing steps (*Grave et al.*, *Kellens et al.*, 2012), on methodological questions concerning data uncertainty and availability (*Müller*

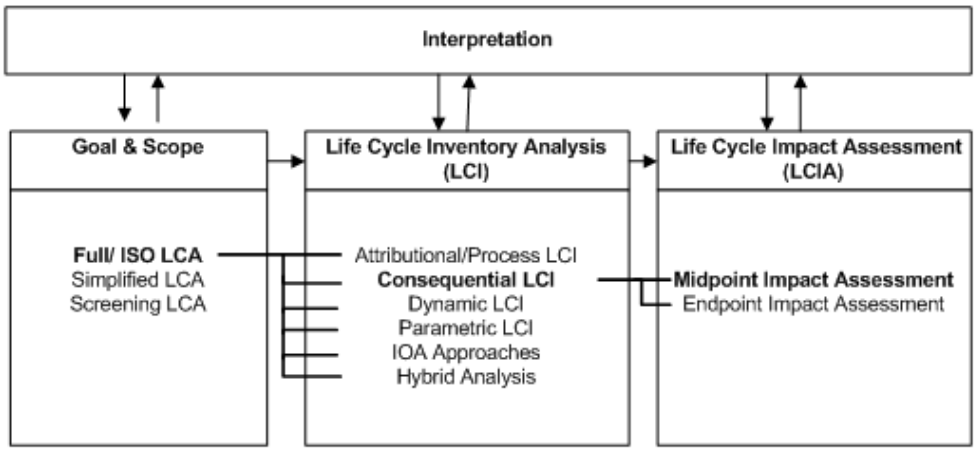

Figure 2.3: Illustration of different LCA methodologies related to the main phases of the LCA (*DIN EN ISO 14044*, 2006).

et al., 2004, *Hübschmann et al.*, 2009) or on the integration of LCA in product design (*Abele*, 2008).

In technology and product development, LCA is applied in process selection, design and optimization (*Azapagic*, 1999b), and therefore multiple methods occurred within LCA such as screening approaches (*Gasafi*, 2006), ecodesign tools and software (*Abele*, 2008) and approaches for combining LCA with optimization methods (*Schmidt*, 2009, *Azapagic*, 1999a).

One challenge for the use of LCA in design and optimization is seen in the mathematical basis of current LCA since optimization problems can not be solved with matrix algebra of the material flow analysis (*Lambrecht*, 2011). Simplified LCA approaches are recommended for early-stage technology development. Hübschmann introduced a simplified LCA for chemical process design, containing three iteratively linked steps: screening, simplification and assessment of reliability (*Hübschmann et al.*, 2009). Simplification means reducing involved life cycle stages and/or processes or impact categories in life cycle impact assessment. However, literature on LCA methodology applied to a case study of a whole technology development process from idea to commercial production is currently not publicly available.

In summary LCA is utilized with very different scopes in technology development but until now a consistent approach that would take into account any given specific requirement is lacking. Such requirements are for example: type of technology development (new or further development), development stage, technology complexity (device development, process development).

2.1.6 Key issues for using LCA

Here, key issues are understood as important criteria within all phases of LCA which shall be clearly defined and be consistent with the intended application (ISO 14044). They should be qualified as important in the frame of the respective methodology according to DIN-ISO (*DIN EN ISO 14044*, 2006, *Klöpffer*, 2009, *Joint Research Center*, 2010, *Zamagni*, 2006). These key issues are: *'intended application', 'reasons for carrying out the study', 'intended audience'*, and *'data requirements'*. Although other issues may be important as well, key issues should be sufficient for comparing different methodologies for the sake of simplicity. To be able to compare different methodologies, the most common answers characteristics to each key issue were identified by a literature study and are listed in Table 2.1.

Key issues	**Characteristics**
Intended Application	- comparison between two or more products - comparison of different designs and/or process chains for one product - steady state analysis of one product
Reasons for carrying out the LCA	- selection - optimization
Intended Audience (stakeholder)	- internal vs. external - technical vs. non technical
Data Quality	- good - medium - poor
Data Type	- quantitative - semi-quantitative - qualitative
Data Source	- measured - calculated - estimated

Table 2.1: Key issues for LCA application.

The ILCD handbook 2010 (*Joint Research Center*, 2010) lists 23 *'intended applications'*, which can be classified into three fields: 1) comparison between two or more products and/or product groups, 2) comparison of different designs and/or process chains for one product, and 3) steady state analysis of one product. *'Reasons for carrying out the study'* are understood as the identification of the decision context whereas the drivers and motivations are neglected. Two decision contexts for using LCA have been identified: selection by comparing and optimizing products or technologies. The answer characteristics for the *'intended audience'* (stakeholder) are classified into professional vs. laymen (technical vs. non-technical) and based on the degree of association to the product (internal vs. external). *'Data requirements'* include three dimensions; data quality, data type, and data source, each representing a separate key issue.

2.2 Micro process engineering

Micro process engineering, a young interdisciplinary field, is connecting physics, chemistry, and engineering to enable new routes in mixing, chemical transformation, and separation in internal structures on a micrometer scale for various applications (Kockmann, 2006). Micro reaction engineering, a part of micro process engineering is the science of chemical conversion in microreactors. In this thesis, the focus is on a microreactor for a highly exothermic heterogeneously catalyzed reaction. Basics about micro reaction engineering are introduced in Section 2.2.1. Section 2.2.2 provides information about differences to conventional reaction engineering in large scale reactors. Within the last Section 2.2.3 the development process of a microreactor is analyzed as a basis for results in Chapter 5.

2.2.1 Micro reaction engineering

2.2.1.1 Overview

Microreactors are characterized by a small interior structure and therefore a high surface-to-volume ratio. This enables excellent heat transfer resulting in improved yield, selectivity, product quality and safety (*Hessel*, 2009). Based on these characteristics application fields are often fast and highly exothermic reactions, such as the Fischer-Tropsch synthesis. Further application fields are heterogeneous multiphase reactions, reactions with low selectivity and productivity in conventional reactors such as stirred-tanks and applications with low production volume e.g. fine chemicals (*Mitropetros and Bazzanella*, 2010). Economic advantages include on-site and on-demand production reducing transportation costs as well as reduced storage and environment protection costs through smaller hold-up of the product in the reactors and on-site.

Figure 2.4 illustrates the interdependencies between the four main issues in microreactor development; these issues are explained in more detail with focus on the considered reactor in this work in the following Sections.

2.2.1.2 Chemical reaction

Microreactors are classified according to the type of chemical reaction and the number of phases involved. Chemical reactions can be distinguished between catalytic and non-catalytic reactions. Homogeneous reactions comprise the class of pure liquid- or gas phase reactions excluding solid-liquid and solid-gas interactions. Heterogeneous reaction systems consist of at least two interacting phases, differentiated into mobile and fixed or conti-

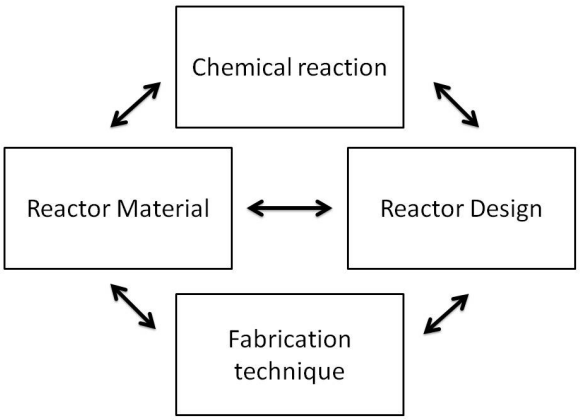

Figure 2.4: Interdependencies in microreactor development (*Kockmann*, 2006, *Pfeifer*, 2012)

nuous and dispersed phase, and are subdivided into three classes: fluid-solid, fluid-fluid, and three-phase reactions. The properties of chemical reactions such as kinetics (reaction rate), thermodynamics (endothermic, exothermic), selectivity, productivity, and hazardous components are influencing the choice of reactor material. The most important factors are temperature resistance, corrosion resistance and thermal properties (*Brandner*, 2006).

2.2.1.3 Reactor Design

The microreactor design is one part within the whole design procedure of the chemical process and plant described in more detail in Section 2.2.3. Compared with other microstructured devices such as micromixers or micro heat exchangers, the microreactor design is more complex due to catalyst integration, as well as the required homogeneous fluid and temperature distribution (*Schirrmeister*, 2006). A microreactor consists of four parts; microreactor core with reaction and cooling channels, flow field distribution structures, flanges/connectors and if necessary pressure vessel as illustrated in Figure 2.5. Besides that, catalyst and catalyst filling or coating is required for catalytic reactions in microreactors.

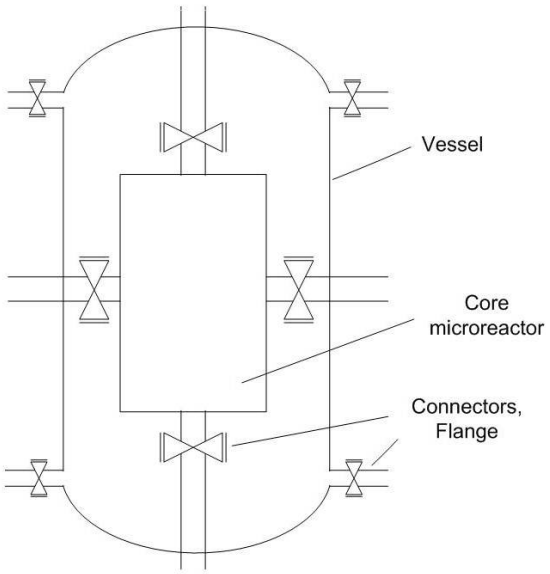

Figure 2.5: Microreactor components (P. Pfeifer, IMVT; personal communication).

Three main design principles for microreactors exist: monolith, membrane and plate design (*Güttel et al.*, 2007, *Hessel et al.*, 2009, *Kockmann*, 2006). Depending on the application and material a microreactor can consists of only one plate or foil. Figure 2.6 illustrates the interior design of a catalytic microreactor. On top of design considerations such as channel diameter and length, wall thickness, catalyst integration method, and cooling or heating design, reaction specific parameters are also of major relevance. The catalyst composition, particle size/coating thickness, catalyst density as well as catalyst pore size/distribution influence the mass and heat transport of the reactants and therefore determine the productivity and product distribution.

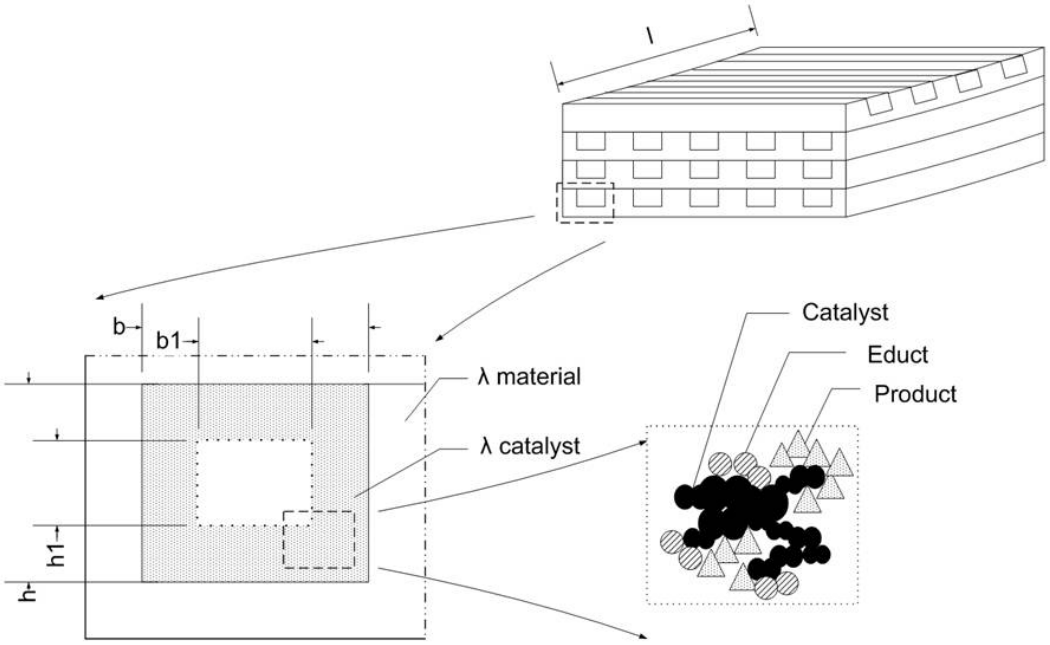

Figure 2.6: Microreactor interior in plate design and the relevant parameters (*Zschieschang et al.*, 2012a) .

2.2.1.4 Manufacturing techniques

Different manufacturing techniques exist for different microreactor parts and materials. Moreover, the reactor design determines the manufacturing technique e.g. in terms of the required precision of the microstructures. Surface roughness, aspect ratio, and channel design are three important parameters for choosing the appropriate manufacturing technique. Manufacturing techniques for each manufacturing stage of a microreactor in plate design are summarized in Table 2.2. In this work only plate design microreactors made from metal foils are considered. This type of microreactor design and chosen material limit the options of manufacturing techniques illustrated in Table 2.2.

Manufacturing step	Manufacturing techniques
Structuring	- mechanical micromachining (milling, turning, drilling) - wet chemical etching - μ-electrical discharge machining - laser ablation - embossing
Assembling	- electron beam welding - laser welding - diffusion bonding - high-Temperature soldering
Packaging	- clamping - thermal shrinking - soldering - welding (Wolfram Inert Gas welding (WIG)), laser or electron-beam)
Catalyst	- filling - coating

Table 2.2: Manufacturing techniques for microreactors in plate design (*Kockmann*, 2006).

2.2.1.5 Reactor material

Today, a variety of reactor materials are available such as metals, plastics, wafer grade silicon, glass or ceramics (*Mills et al.*, 2007) . The application field, chemical reaction, and manufacturing techniques determine the choice of material. Figure 2.7 illustrates some major interdependencies in the decision processes from pre-design to design to reactor manufacturing.

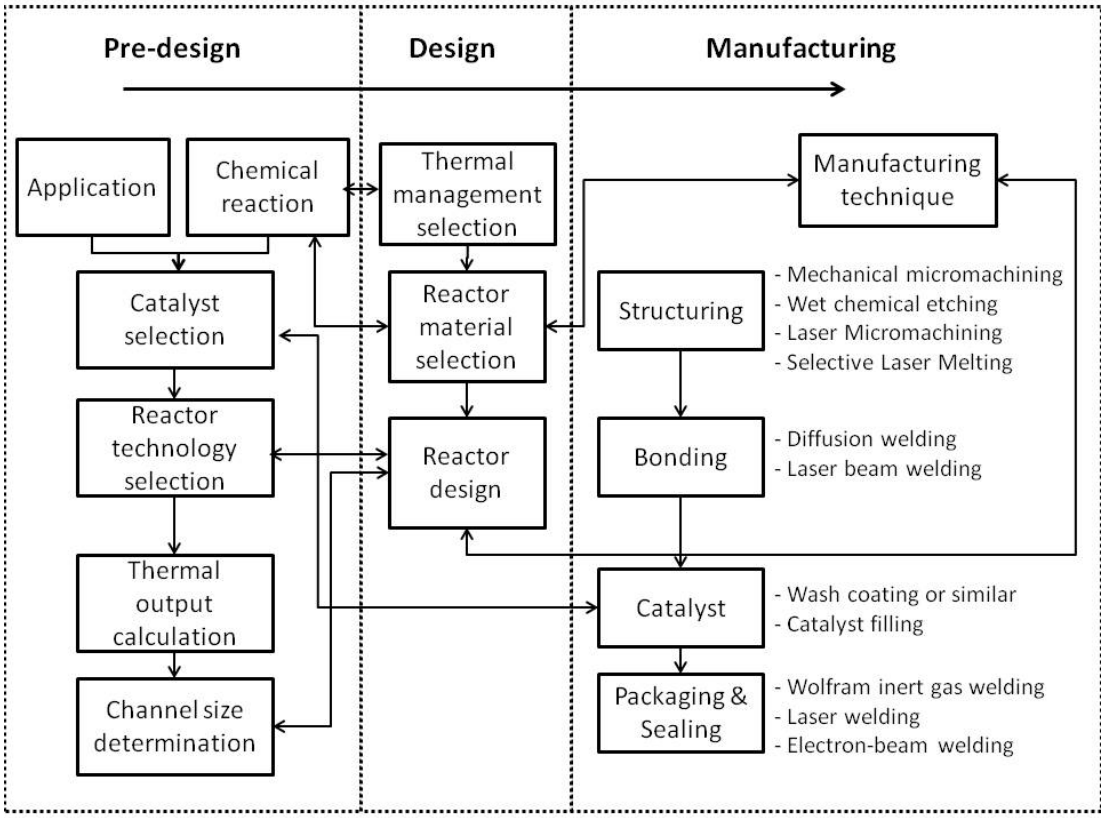

Figure 2.7: Interdependencies in the microreactor development process from pre-design to manufacturing (*Zschieschang et al.*, 2012a).

2.2.2 Paradigm shift and novel process windows

The paradigm shift in chemical process and plant design is realized through the application of continuous processes in micro reaction engineering compared to batch processes in conventional process engineering. Advantages of micro reaction engineering such as enhanced heat and mass transfer result in process intensification and allow controllable reactions under harsh conditions hitherto un-feasible with the 'old' vessel operations. This process intensification via flow or continuous chemistry with microreactors is also called NPW (*Hessel*, 2009).

Major strategies and process intensifications of NPW are: new chemical transformations, routes at elevated temperature and pressure, routes at increased concentration or even solvent free, process integration and simplification, and routes in the explosive or thermal runaway regime (*Hessel*, 2009).

This new pathways require a hand-in-hand design of the devices (microreactors) and processes for the considered chemical reaction (*Hessel*, 2009, *Charpentier et al.*, 2012).

2.2.3 Technology development

2.2.3.1 Technology development phases

The design process of new micro reaction engineering processes involves the development of new chemical synthesis routes and the development of the reactors (devices), processes and plants thereof. In this work the focus is on the design of the microstructured reactors for a specific chemical reaction, the Fischer-Tropsch synthesis. New chemical synthesis routes and project management in chemical synthesis routes, reactor, process and plant design are neglected.

In principle, the design process of the new reactor consists of three phases: idea, engineering and production. Engineering is further divided into basic and detailed engineering and production in prototype and industrial production. In contrast to conventional technology, the cost structure shows special attributes in microsystems development which covers micro reaction engineering (*Kasper*, 2000). The prototype production is very time and cost intensive, and therefore proper simulations of designs, processes and systems are required in advance to reduce the needed number of prototypes and therefore reduce time and costs. Figure 2.8 shows the course of three important technology development parameters during individual development phases.

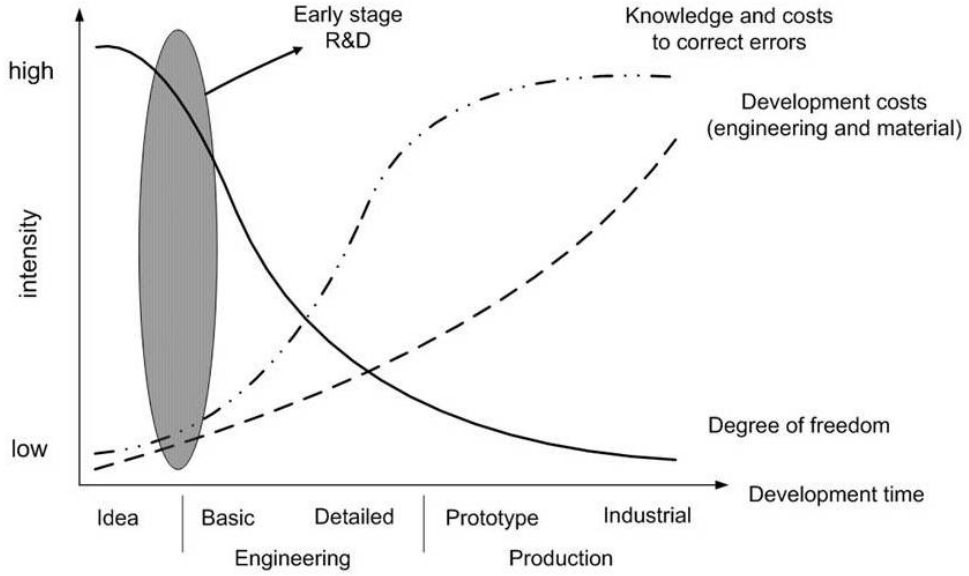

Figure 2.8: Course of three important technology development parameters during individual technology development phases (*Schirrmeister*, 2006).

Within the development phases, simulation of processes and plants combined with established scale-up/numbering-up procedures are of major relevance in terms of reduced time-to-market intervals in the future. Figure 2.9 illustrates the comparison between an established reaction engineering approach and the time-to-market phases of plants including microreactors (*Schirrmeister*, 2006).

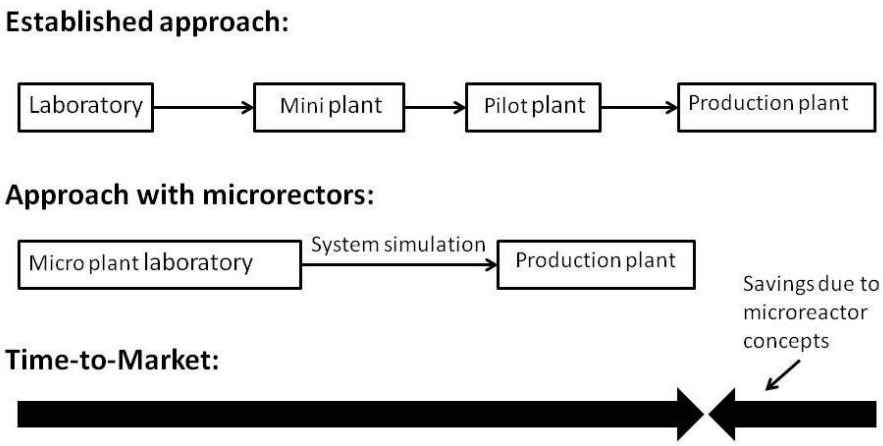

Figure 2.9: Time-to-market of processes and products for established approaches and approaches including microreactors (*Kockmann*, 2006).

2.2.3.2 Stakeholders in technology development

As described in the previous section, the development of technologies consists of different stages classified in terms of knowledge and degree of freedom. Within each developmental stage, different stakeholders and decision makers are involved. These could be the society, investors in general, a company, the head of R&D or responsible engineers.

Engineers are involved in early R&D as they have to find the most appropriate solution within defined boundary conditions determined by the head of R&D or decision makers in industry. The given boundary conditions itself depend on the level of technology development, where level is associated with complexity. This means for example, the development of the microreactor (device) is on a lower level than the process, because the process consists of the microreactor and other devices and is therefore more complex.

Due to different involved stakeholders and developmental stages of technology, there exist multiple interests, such as reactor design with the lowest energy consumption (engineers) and costs for carbon dioxide prevention by the utilization of the microreactor. The complex network of dependencies and interactions is illustrated in Figure 2.10.

The involved stakeholders for the microreactor development are often engineers and the

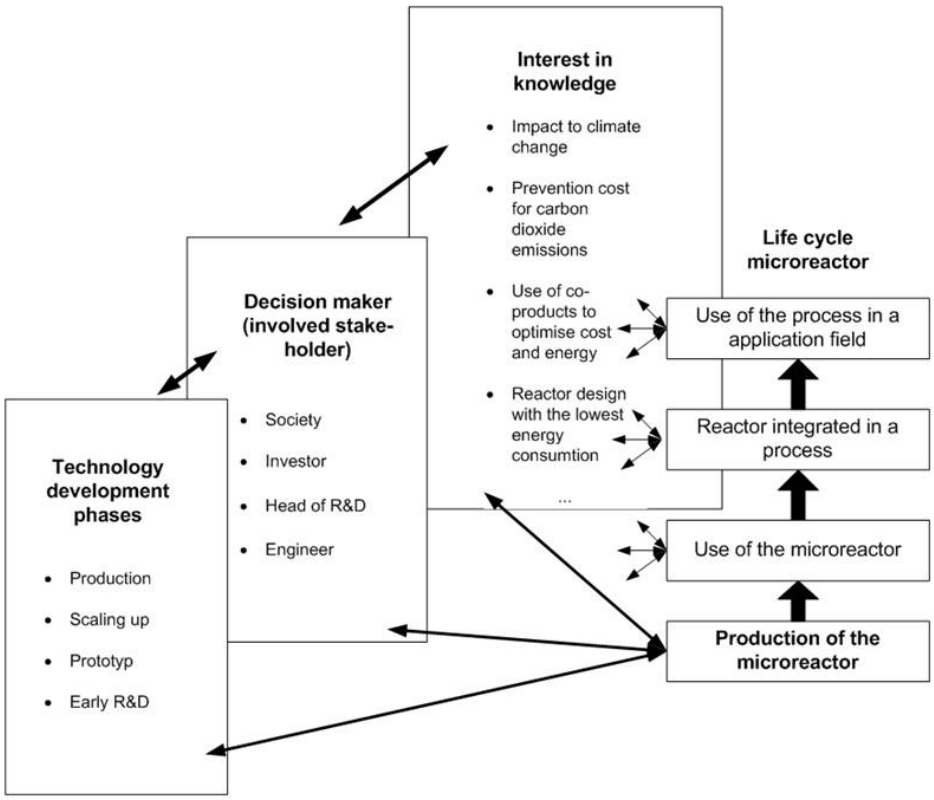

Figure 2.10: Dependencies between the technology development phase, decision maker, interests and the life cycle of a microreactor.

head of R&D. Engineers involved in the reactor development are mainly interested in 1) energy- and raw material consumption according to the reactor design and 2) the identification of specific parameters in manufacturing, which are improvable or can not be attributed to energy- and raw material consumption (P.Pfeifer, M. Kraut IMVT; personal communication). The interests from the head of R&D could be the integration of the microreactor in a process for a specific application field.

2.2.3.3 Data types occurring in technology development phases

The development of technologies is not only a complex system, consisting of different development stages, stakeholders and interests. It is as well characterized by huge differences in existing knowledge and therefore data availability is a major issue. Here, the term data has to be differentiated from the three basic items - material, energy and information - which describe processes and conditions in nature.

Focusing on the stakeholder engineer and their interest as stated in Section 2.2.3.2, in

general, generic data are often missing for design depending manufacturing processes. Two main reasons are identified for this gap. First, manufacturing is usually depending on the design and therefore it is not possible to have experimental data (energy and material consumption) in advance for a new design produced. Using comparable, already known processes is limited due to the fact, that the combination between the design parameters and process specific manufacturing data is usually not foreseen a priori. Second, the difference in the data type is large, because the data obtained in the design of the microreactor are of different nature compared to material and energy consumption in the manufacturing process. Both data sets are connected by more or less complex functionalities between specific manufacturing and design parameters of the reactor resulting in an overall material and energy consumption.

Figure 2.11 illustrates the interdependency between three relevant aspects in technology development - specific parameters (*P*), boundary conditions (*B*) and requirements for the technical system (*R*) - in terms of data type and requirement. The aim is to illustrate the complex interactions and networks for decision making (*D*) in early Research and Development (R&D). A specific parameter stands for one degree of freedom and is used for any kind of parameter in technical context, such as material properties (heat conductivity), temperature, or design parameters (channel width). Requirements are understood as a specification of the developed technical component, given by the user of the technology. The third part is the interface between engineer and user, the so called boundary condition. Those could be maximum size of channels inside a microreactor determined from plate thickness.

All three components - parameter, boundary conditions and requirements - can be either independent from each other or dependent on each other. For example, the catalyst productivity (catalyst property) is independent from catalyst void fraction (catalyst volume per channel volume), but only within a specific bandwidth. Independence of specific parameters is thus a matter of the chosen bandwidth.

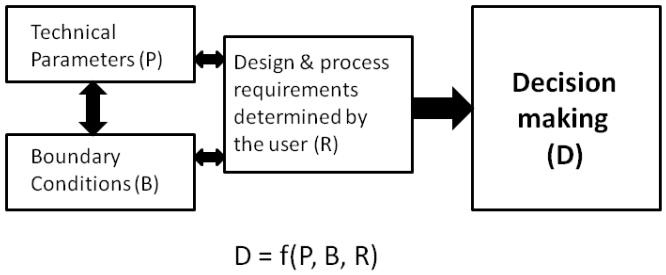

Figure 2.11: Dependencies of specific parameters (*P*), boundary conditions (*B*), design and process requirements (*R*) and the decision making (*D*).

2.3 Case study

This chapter introduces the case study of an offshore GtL process with the Fischer-Tropsch synthesis in micro reaction engineering which is analyzed in this thesis. Within the following Sections, basics in chemical synthesis routes of the Fischer-Tropsch synthesis are introduced; the chemical concepts for FTS in Section 2.3.2, the catalyst development in Section 2.3.3 the process of syngas production in Section 2.3.4 and the GtL process in Section 2.3.5. The final Section 2.3.6 provides an overview on application fields and potentials for FT microreactors and small scale GtL plants.

2.3.1 Introduction

Although Fischer-Tropsch technology is an established technology, the application in micro reaction engineering is quite new. This is due to the fact that the issue of gas flaring became interesting for the conversion of small amounts of so called stranded gas, e.g. by the Global Gas Flaring Reduction (GGFR) initiative (*The World Bank*, 2002). Stranded or associated gas is a byproduct of crude oil production that is currently not refined further in a profitable manner, due to the specific boundary conditions of the platform location and the small amount of the byproduct.

Feasibility studies were undertaken for highly exothermic gas-phase reactions in fixed-bed reactors (*Güttel and Turek*, 2009b, *Güttel et al.*, 2007) and the comparison of different reactor types for FTS by a simulation study (*Güttel and Turek*, 2009a) was conducted recently. Within recent years, a first FTS test reactor, a multichannel packed bed microstructured reactor (*Myrstad et al.*, 2009), and prototype reactors (*Deshmukh*, 2010, *Tonkovich*, 2011) were developed. Further developments have been undertaken in the field of synthesis gas (syngas) production; methane reforming (*Tonkovich*, 2004), steam reforming (*Kolb*, 2009) and partial oxidation in microreactors (*Pfeifer*, 2009). However, until now no prototype 'micro' GtL plant exists.

Subject of investigation in this thesis is the development of a microreactor for Gas-to-Liquid fuel conversion by applying Fischer-Tropsch synthesis. The research is conducted at the KIT at the IMVT. The microreactor for FTS is part of the GtL process system, which includes synthesis gas production besides FTS. Until now, there exist only GtL plants based on conventional fixed bed, fluidized bed or slurry type reactor.

The development of a microreactor for the FTS at IMVT, KIT is in early stage of R&D. The proof of concept for the FTS in a microrstructured test reactor was successfully realized together with co-operation partners and first results were published in (*Myrstad et al.*, 2009). As a next step, a FTS microreactor with higher throughput per hour will be built.

A reformer microreactor for the syngas production may be set up as the main components of a GtL microplant. One of the most promising application fields is the use on offshore oil-platforms in remote areas. Since data exists only from a single test-reactor, no data are available for different design options of the FTS reactor and the GtL process. Moreover, also the catalyst development is still ongoing.

2.3.2 Chemical concepts

The FT synthesis is a strongly exothermic heterogeneous catalytic reaction (reaction enthalpy $\Delta_R H^0$ (250 °C) = - 158.5 kJ mol^{-1} CO (*Chemgaroo*, 2012)), that converts carbon monoxide and hydrogen into a mixture of liquid hydrocarbons (mainly linear and saturated C-H chains).

Stoichiometry

The chemistry and kinetics behind the Fischer-Tropsch synthesis is complex and until now object of intensive research and controversial discussion (*Schulz*, 1999, *Iglesia*, 1991, *Steynberg*, 2006). Nevertheless, the complex reaction network can be simplified into the following four reactions with the products: methane (Equation 2.1), heavier hydrocarbons (Equation 2.2), alcohols (Equation 2.3), and carbon dioxide (Equation 2.4). The reaction in Equation 2.4 is called Water-Gas-Shift (WGS) reaction.

$$CO + 3\ H_2 \longrightarrow CH_4 + H_2O \tag{2.1}$$

$$n\ CO + (2n+1)H_2 \longrightarrow C_nH_{2n+2} + n\ H_2O \tag{2.2}$$

$$n\ CO + 2n\ H_2 \longrightarrow C_nH_{2n+2}O + (n\text{-}1)\ H_2O \tag{2.3}$$

$$CO + H_2O \rightleftharpoons CO_2 + H_2 \tag{2.4}$$

Hydrocarbon products from the Fischer-Tropsch process can be classified in fuel gas (methane, ethane), Liquefied Petroleum Gas (LPG), and a mixture of gasoline, diesel, and wax summarized as a fraction with more than 5 carbon atoms (C_{5+}).

Operating conditions

In addition to the type of catalyst, type and amount of promoter, feed gas composition and type of reactor used, operating conditions of the FTS such as operating pressure, operating temperature, and gas hour space velocity have a great influence on the conversion

rate and the product distribution of the FTS (*Steynberg*, 2006). However the dependency is too complex to be explained here in brief. Simplified assumptions have been made to see the influence on the LCA.

Product distribution

In principle, the product distribution is calculated either with the Anderson-Schulz-Flory (ASF) distribution, a simplified model of FTS reducing affected parameters to only one, the so called *chain growth probability* (*Flory*, 1936, *Puskas*, 2003), or with more sophisticated models. Based on the simplified models, the sophisticated models apply further parameters to calculate the product distribution for example olefin re-adsorption, incorporation, hydrogenation and isomerisation (*Schulz*, 1999). The chain growth probability is widely used in practice and therefore several data sets are available. Kinetically, reasonable uncertainty remains mainly regarding the dependency of the chain growth probability on the chain length for lower molecular mass hydrocarbons in practice since the kinetic environment can not be kept identical (*Puskas*, 2003). In (*Tavakoli*, 2008) it is indicated that the ASF distribution is not suitable for nanoparticle catalyst systems. Equation 2.5 shows the product distribution described by the chain growth probability, where W_i is the weight fraction of the products, i denotes the carbon number of the produced molecule and α the chain growth probability.

$$W_i = i(1-\alpha)^2\alpha^{i-1} \tag{2.5}$$

2.3.3 Catalysts development

Catalyst development is driven by the chemical reaction involved and the technical application. Technical application in this context means the reactor type used for FTS (slurry phase reactor, fixed-bed reactor, microreactor). The catalyst production for FTS in commercial reactors is carried out either by melting of the catalyst components, chemical precipitation from metallic salt solution or carrier material impregnation with metallic salt solutions (*Chemgaroo*, 2012). Appropriate catalyst metals are iron, cobalt, nickel and ruthenium (*Steynberg*, 2006). Nickel was abandoned due to the high methane concentration in the product (*Steynberg*, 2006). Ruthenium is only used as promoter not as main catalytic active species because of the high costs. This leaves cobalt and iron as main components for FTS, whereas cobalt is used for Low Temperature Fischer-Tropsch synthesis (LTFT) and iron for High Temperature Fischer-Tropsch synthesis (HTFT) (*Steynberg*, 2006, *Güttel et al.*, 2007). LTFT produces mainly hydrocarbons with more than twenty carbon atoms

in the molecule (wax), whereas HTFT produces lower hydrocarbons with mainly three to eleven carbon atoms in the molecule. As mentioned above, the used technology (reactor type) also influences the catalyst development in terms of catalyst composition, particle size, required activity, and lifetime of the catalyst. Until now, cobalt catalyst show high potential for LTFT synthesis in microreactors (*Myrstad et al.*, 2009, *Deshmukh*, 2010).

2.3.4 Synthesis gas production

Syngas, the feed for FTS, consists of hydrogen and carbon monoxide. Syngas production is the most expensive part of the the GtL process (*van Vliet*, 2009). According to the feedstock, different technologies are available for the syngas production, but since the focus in this work is on associated gas as feedstock other technologies than reforming natural gas are neglected. Most common technologies for syngas production from associated gas are steam reforming and partial oxidation or autothermal reforming the combination of both. Prior to syngas production, gas treating technologies are used to avoid adverse effects in downstream processes by reducing higher hydrocarbons (Natural Gas Liquids (NGL), LPG), carbon dioxide and sulfur components (*Steynberg*, 2006). Until now, microreactors for syngas production are not commercial available.

2.3.5 Gas-to-Liquid process

The Gas-to-Liquid process consists of six main parts illustrated in Figure 2.12. Each part consists of one or more subprocesses. The desired product distribution or product slate determine the choice of FT synthesis (HTFT, LTFT, iron or cobalt-based catalyst) and therefore the operating conditions in the FTS unit. This will in turn determine the desired characteristics of the syngas. The feedstock characteristic will then, together with the desired product gas, determine the syngas technology (reforming, gasification, Autothermal Reforming (ATR)) to be applied. In conclusion, there are various design options for the GtL process with interdependencies between the single processes.

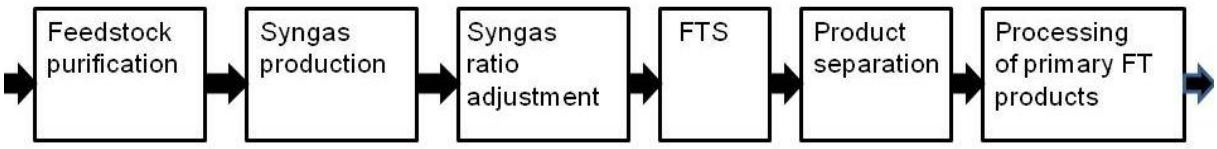

Figure 2.12: GtL process chain with main components.

2.3.6 Application fields and potentials

The focus lies on FTS in microreactors, and therefore only relevant application fields and potentials for this type of technology are considered. Microreactors for FTS are reported to be useful for small decentralized plants (*Tonkovich*, 2004), whereas large scale central plants for large natural gas fields are equipped with conventional reactors such as the Sasol slurry phase reactor for LTFT or with advanced techniques like the Synthol® reactor for HTFT (*Heydenrich*, 2005, *Steynberg*, 2006).

With regard to the feedstock for the synthesis gas production, biomass and associated gas are possible inputs for the microreactor. Biomass as feedstock is however considered to be outside the scope of this thesis. Associated or tail gas occurs as disposal during oil production and processing. Gas flaring is widely used as disposal method of methane to reduce global warming and for safety reasons, especially in areas with insufficient infrastructure for the use of this gas (primarily methane). Global warming is reduced due to the lower GWP of CO_2 which is 1 kg CO_2-Eq. per kg CO_2 gas compared to 21 kg CO_2-Eq. per kg CH_4. The World Bank in cooperation with the Government in Norway launched a GGFR initiative in 2002 with the goal to eliminate gas flaring and venting (*The World Bank*, 2002, *GFRP*, 2004). Since data on the gas volume and flaring places are not provided by the oil companies, GGFR sponsored a project which investigated the use of satellite data for the gas flaring detection and estimation of the volume. As a result, gas flaring volumes and places could be estimated for individual countries. In 2008, worldwide gas flaring is equivalent to 21 % (139 billion cubic meters (BCM)) of the natural gas consumption of the USA (*Elvidge*, 2009). The most relevant result of this project for the application of microreactors is, that most of the gas flaring occurs at remote oil production or processing facilities, many of which are offshore.

With respect to the promising application field of the GtL plant build with microreactors for substitute gas flaring, the overall aim in the microreactor development is to develop a technology with as low CO_2 emissions as possible.

2.4 Challenges for LCA in microreactor development

The aim of this thesis is to examine how current state of the art LCA can be applied in early R&D of technologies on the example of a microreactor for a GtL process.

The significance of LCA in technology development is greatest at very early stages in the development process. This is largely due to the fact that the early R&D phase provides the highest degree of freedom regarding the selection of technology design parameters. Technologies are developed for a special application and defined by required performance specifications. To achieve these, several design alternatives may exist. In this context, the goal of a timely LCA is to gain information about the environmental impact of different design alternatives in order to enable the identification of optimized design variations. It is important to note that such optimization has occur on a system level rather than on the level of individual components.

Microreactor development belongs to a type of technology, where the performance of a complex system - GtL - depends on a major component - the microreactor for FTS-, and therefore the design of the microreactor influences the GtL system performance. In addition, the chemical synthesis applied in the microreactor strongly affects the microreator design itself. Richard W. Hamming (*Hamming*, 1997) summaries the main challenge within systems design as follows:

'If you optimize the component you will probably ruin the system performance'

Importantly, LCA for the development of a microreactor driven GtL process requires the ability to model GtL process performance, taking into account several different FTS microreactor variations.

The microreactor development process exhibits the following main characteristics that are relevant for the application of the LCA methodology:

- Varying system boundaries
- Varying involved stakeholders and stakeholder interests
- Interdependency between technical equipment (catalyst, chemical reaction, microreactor, process) developed in different development phases
- Use of multiple modeling techniques with high level of detail

- Limited data availability in terms of material flows in foreground process (due to the not-yet existing technology)

Derived from above mentioned characteristics and the requirements for the application of the LCA stated in Section 2.1.6 two research questions arise:

1. How to model the whole microreactor development?
2. How to apply LCA to this model?

The answer to the first question must include solutions for time dependent system complexity and changing system boundaries of the microreactor development model. In particular, a consistent modeling framework is required to allow analysis of multiple individual types of microreactor models in each single development phase. The solution to the second question requires a method enabling to analyze different modeling concepts in LCI with the aim to realize a in- and output material flow analysis.

CHAPTER 3

Systems analysis of complex systems

Within this chapter the basic theoretical concepts of general systems theory and two applications for complex systems are introduced with the aim to provide background information to answer the question on how to model microreactor development and on how to apply LCA to this model.

3.1 General systems theory

3.1.1 Basics

Systems analysis is the science of describing and analyzing systems. Although the systems might be of different nature, the simplified general procedure for systems analysis is the same for all, containing system description, modeling and model analysis as illustrated in Figure 3.1.

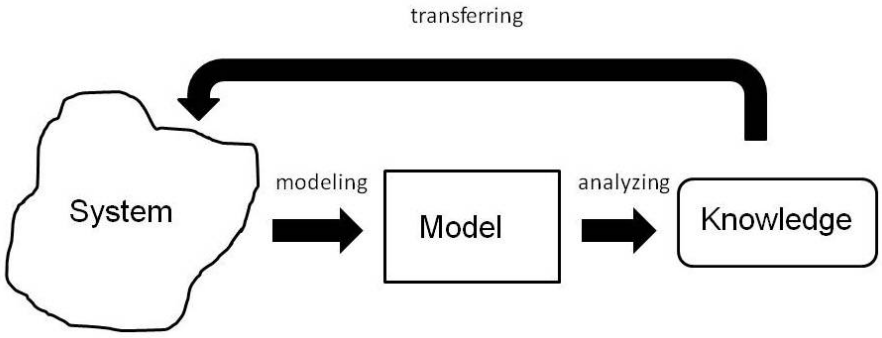

Figure 3.1: General procedure of the system analysis (*Häuslein*, 2004).

The general procedure to setup a model requires first the decomposition of the system in subsystems, second the definition of single modules and third the compositions of the modules to the final model (*Häuslein*, 2004). There exist four decomposition strategies for systems which are function-, data-, event-, and object-oriented decomposition. Systems analysis methodologies are selected according to the chosen decomposition strategies, i.e. object-oriented decomposition requires object-oriented analysis tools.

3.1.2 System description

A system (S) is described by the system structure, defined as organization of in itself closed elements and their relation to each other, and the system boundary. In accordance to the in- and outputs between the system and the surrounding area, two system types are distinguished: the open and closed systems. The open system has in- and outputs to the surrounding area while the closed system has none.

The system complexity is based on three distinctions made with regard to the elements: 1) *number*, 2) *species* and 3) *relation between the elements* (*van Bertalanffy*, 2001). The species of an element can be for example a natural number, a molecule or a microreactor. By neglecting the element species, Equation 3.1 describes the complexity as the quotient between the number of relations (n_R) and the number of elements (n_E) of the considered system.

$$K = \frac{n_R}{n_E} \tag{3.1}$$

Thus, a small number of elements of one species in isolation (summative system) is less complex than a small number of elements of one species in multiple relation to each other (constitutive system). When describing a system, the level of detailing determines the level of specification. Therefore, if the considered system is subdivided in subsystems ($S^{'}$), it might be necessary to further subdivide into further subsubsystems depending on the level of detailing. A formal system definition is described by fourteen single definitions, see Appendix A, and is illustrated in Figure 3.2.

In contrast to the above mentioned formal system definition, three different system concepts can be distinguished in practice: the hierarchical, functional and structural concept, each focusing on one aspect of the general systems theory illustrated in Figure 3.3.

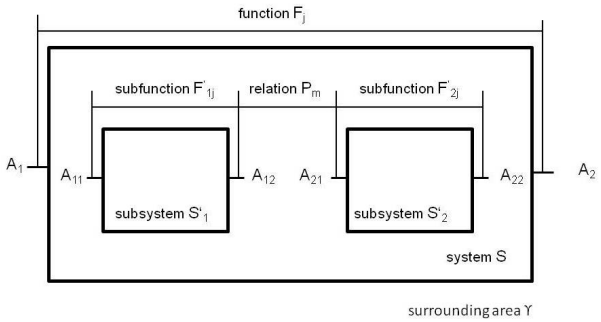

Figure 3.2: Schematic representation of the formal system definition (*Ropohl*, 1979). Variables: system (S), in- and output attribute (A), system function (F), relation between systems (P).

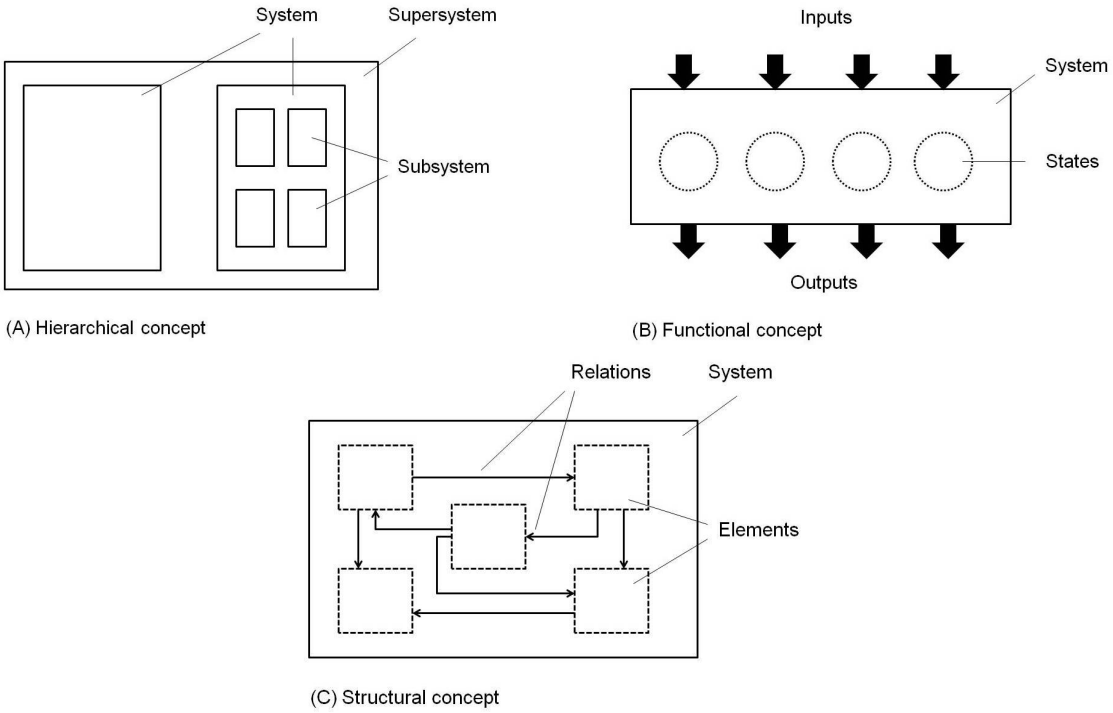

Figure 3.3: System concepts in practice (*Ropohl*, 1979, 2009).

Besides the formal description, systems are categorized according to their properties facilitating the categorization of analysis methodologies. System properties are for example change in state (dynamic, static), level of existence (artificial, real), predictability of properties (deterministic, stochastic) and level of concretion.

3.1.3 Modeling

Models are earmarked transformations of systems, which means they strongly depend on the proposed interests of the systems analysis (*Niemeyer*, 1977). Based on that transformation, system terms and definitions are also adaptive.

Modeling has two main characteristics. First, modeling is strongly influenced by the aim of the systems analysis. This characteristic of a model is called *shortening attribute* (*Stachowiak*, 1973). Models could be either appropriate or not-appropriate, but never right or wrong models.

The second characteristic is called *transformation attribute* (*Stachowiak*, 1973). The transformation attribute describes the mapping between system and model elements. Isomorph mapping assigns to each system element one model element whereas homeomorph mapping assigns to each model element one system element. In the second case, some system elements might be unassigned.

Due to the strong relation between the aim of the systems analysis and the modeling, validation of the model is limited to this specific aim and needs modification when changing the aim.

In principle models are aimed to describe, explain, forecast, design or optimize a system.

3.2 Sensitivity model

The *sensitivity model* by Frederic Vester is a procedure on how to analyze complex systems with a biocybernetic approach including system description, modeling and systems analysis (*Vester*, 2007, 2011). The biocybernetic approach is a new way of describing a system and the functional interdependencies between the subsystems by looking from inside towards the system characteristics instead from outside. The full set of characteristics of this approach are:

- Steering from inside the system
- Steering is limited to the support of self regulations
- Indirect problem solving
- Long-term stabilization of system dynamic through flexibility

The aim of using bio-cybernetical approaches are to make the system *'as viable as possible in terms of self-regulation, flexibility, and controllability'* (*Vester*, 2007, p.185). Based on this, a three level process with nine interconnecting operations containing multiple methodologies is to be applied within the sensitivity model.

1. Description
    - Describing the system
2. Recognition
    - Registering actuating variables
    - Checking variables for systemic relevance
    - Studying interactions
    - Determining the role of variables and interactions within the system
    - Examining overall interconnectedness
3. Analysis
    - Cybernetics of individual scenarios
    - 'What-if' forecast and policy test
    - Evaluating the system and formulating strategy

The structure of the sensitivity model is recursive. This means, that each operation stage remains open and therefore allows permanently updating of the system. The sensitivity model is widely used for large-scale systemic studies and projects in the field of corporate strategic planing, technological assessment, development-aid projects, security policy and conflict analysis, traffic and urban planning. One example for urban and regional planning is the Cooperative Ecological Research Project (CERP) for the Tianjin region in China in collaboration with the UNESCO program *Man and the biosphere*.

3.3 Object-oriented software engineering

Object-oriented software engineering as one of the most flexible methods for highly sophisticated and complex systems design has become increasingly important during the last twenty years. Conventional models and approaches for software system design could not achieve the requirements of the more complex systems and the challenges of the new application fields (*Heinrich*, 2008).

These challenges are for example the development of systems with multi level architecture, client-server and distributed systems. The increasing complexity due to incorporating multiple participants (stakeholders) and more complex technical systems as well as the need for continuously adaptive models generate requirements towards software engineering.

Based on these challenges, requirements towards the new methodology are an integrated approach, inheritance, continuous concepts and modeling and improved possibilities for changing, maintaining and extending the system. Object-oriented software engineering provides multiple concepts and principles to achieve the above mentioned requirements such as class and object, encapsulation, modularity, polymorphism, and messaging. The basic principles are introduced briefly in the following paragraphs. (*Häuslein*, 2004, *Heinrich*, 2008, *Garrido*, 2009)

Object

Objects are defined by their attributes and methods. Although having the same attributes and showing the same behavior than other objects, each object has its own identity. Attribute access is only allowed by the methods provided. The communication with other objects is carried out by defined interfaces.

Class

A class defines the structure and the behavior of the objects in that class and consists of attributes and methods. Attributes in a class are defined as data declaration, and methods as functions on how to operate with the attributes.

Encapsulation

Attributes of a class are only available by the methods provided by the class. Therefore, methods and attributes of a class belong together. This is called encapsulation or secret principle.

Modularization

Dividing a system into smaller systems which are easier to analyze and solve is a common approach for complex systems. This partitioning of a system into smaller systems is also known as decomposition and the small parts are called modules. In object-oriented design, classes are decomposition units (modules) of the whole system.

Messaging

The communication between objects takes place through messaging. A common method for the implementation is the Client-Server principle.

Polymorphism

Classes from different levels in hierarchy understand the same messaging although the methods for the answer of the messaging might be completely different. This is called polymorphism.

CHAPTER 4

New methodology for applying LCA in technology development

This chapter provides the new methodology for using LCA in early stage R&D of technology development on the example of microreactor development. The first Section 4.1 describes the microreactor development model using the general systems theory as modeling framework. Based on this, the second Section 4.2 summarizes the requirements towards the new methodology, which is then presented in the third Section 4.3. Section 4.4 explains the whole work flow for the new methodology and finally, Section 4.5 introduces new implementations necessary for applying this methodology on the software-level.

4.1 Simplified microreactor development model

Within Section 4.1.1 a model of the whole microreactor development is created based on the general systems theory. In order to represent the changing system boundaries and system complexity within the development time, one axiom of the general system theory was adapted as explained in Section 4.1.2.

4.1.1 Modeling framework

The microreactor development as described in Section 2.2 - 2.3 is a very complex system. Modeling this type of system has two main challenges. The first challenge relates to the number and the type of modeling techniques used within microreactor development. Microreactor development involves multiple modeling techniques which are strongly different from each other in terms of level of detail, necessary data about the system to be analyzed and interests (involved stakeholders). Since the modeling techniques in microreactor development are established and therefore determined, a more general modeling framework

should be investigated for the whole microreactor development allowing the use of those single modeling techniques while providing a consistent framework for all.

The second challenge relates to the time-dependency of the single microreactor development phases and therefore the system boundaries. Starting with a single technical component (catalyst, microreactor) on a micro-level, the development continue with the integration of the single components in a process (FTS, GtL) on a meso-level and finally ends on the macro-level with the process design for a special application (offshore GtL application).

According to the above mentioned challenges - system complexity and time dependent system boundaries - two basic concepts of the general systems theory are used for modeling the microreactor development.

The first concept of the modularization is used to reduce the system complexity by braking down the whole system into single parts called modules with defined system boundaries. These system boundaries are determined according to the stakeholders interests. The implementation of hierarchies as second concept was chosen for two reasons. First, to illustrate the interdependencies between the developed technical components within different development phases and second, to illustrate the interdependencies between the technical components within one development phase. Based on these two concepts, a simplified model of the microreactor development was created, illustrated in Figure 4.1.

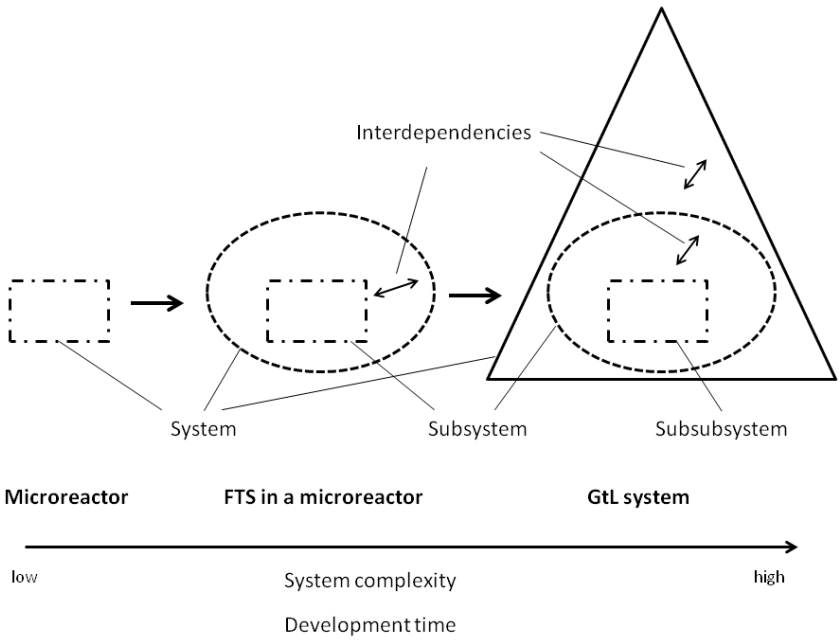

Figure 4.1: Simplified microreactor development model.

Here, *Microreactor, FTS in a microreactor* and *GtL system* each representing one development phase as well as the technical component under investigation in the development phase.

4.1.2 Interdependencies in the general systems theory

4.1.2.1 Interconnections within the same system hierarchy

The general systems theory provides a definition for interdependencies between systems within the same system hierarchy illustrated in Figure 4.2 and Equation 4.1. In order to realize the interdependencies between and within the development phases of the technology (microreactor), the Definition requires extension.

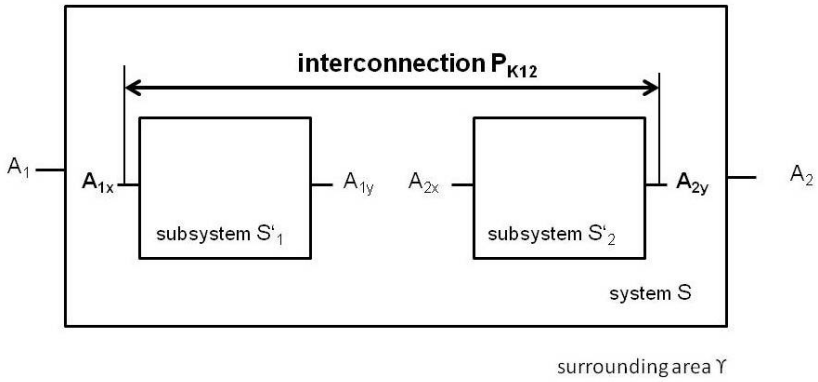

Figure 4.2: Graphical illustration of the interconnection P_{K12} within the same system hierarchy.

$$P_{K12} \subset A'_{1x} x A'_{2y} \tag{4.1}$$

with $a'_{1xn} = a'_{2yn}$ for all n

If A'_{1x} is the input of subsystem S'_1 and A'_{2y} the output of subsystem S'_2 than the double-digit identity relation between A'_{1x} and A'_{2y} is called interconnection P_{K12}.

4.1.2.2 Interconnections between different system hierarchies

In order to illustrate the interdependencies between the modules occurring in different system hierarchies, the definition from Ropohl (*Ropohl*, 1979, 2009) was developed further as shown in Figure 4.3.

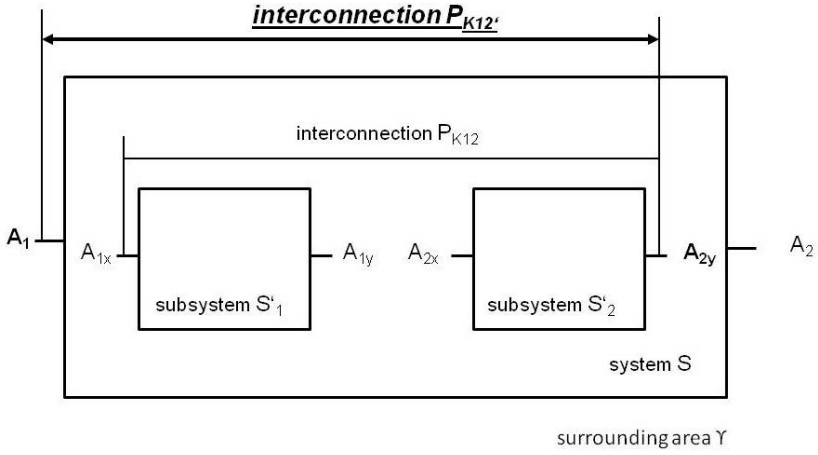

Figure 4.3: Graphical illustration of the interconnection $P_{K12'}$ between different system hierarchies.

Equation 4.2 combines the output of a module in a lower system hierarchy (subsystem) with the input of a module in a higher system hierarchy (system). The interdependencies between the technical system performance (GtL system) and system components (FTS microreactor) as described in Section 2.3 can thus be modeled by using the general systems theory.

$$P_{K12'} \subset A_1 x A'_{2y} \tag{4.2}$$

with $a_{1n} = a'_{2yn}$ for all n

If A'_{2y} is the output of subsystem S'_2 and A_1 the input of system S_1 than the double-digit identity relation between A'_{2y} and A_1 is called interconnection between system hierarchies $P_{K12'}$.

4.2 Requirements towards the new methodology

Applying LCA to microreactor development poses two main challenges. The first challenge relates to the modeling of the considered system. Actually, modeling of a given system from a LCA or a R&D (here used for microreactor development) perspective may result in divergent models of this system, as illustrated in Figure 4.4. This is most likely due to differing decomposition strategies used in LCA and R&D modeling resulting from different interests of the involved stakeholders. The strong relationship between stakeholder interest and model result is one basic axiom of the general systems theory (*Ropohl*, 1979, *Bossel*, 1994). Thus, LCA analysis results cannot simply be transfered to the R&D model. According to the basic axiom of the system analysis, the application of LCA to identify environmental hot spots in microreactor development seems incompatible.

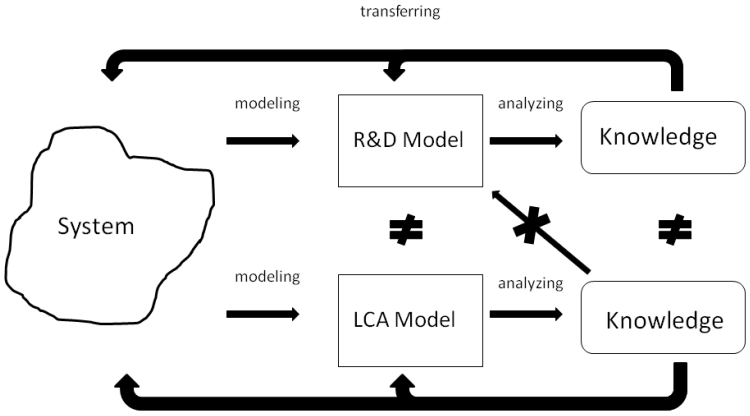

Figure 4.4: Differences in modeling a given system in LCA and R&D.

The second challenge concerns the time dependent system complexity and interconnections between the development phases. Although modeling the technology development is feasible with the expanded general systems theory definition (see Section 4.1.2), the application of LCA remains a challenge due to the requirements within the LCA. Reasons for this are incompatibilities between LCA and R&D models with respect to key issues for using the LCA methodology, in particular within the goal and scope definition and LCI as introduced in Section 2.1.6. According to the goal and scope definition, LCA is required for each development phase because of different stakeholder interests and system boundaries. In addition, the material flow analysis is required in LCI, whereas various other analysis methods focusing on different data types are required in R&D. Finally, there are two time lines to be considered, the technology development itself and the life cycle of the components of each development phase. Figure 4.5 illustrates the two time lines of the microreactor development.

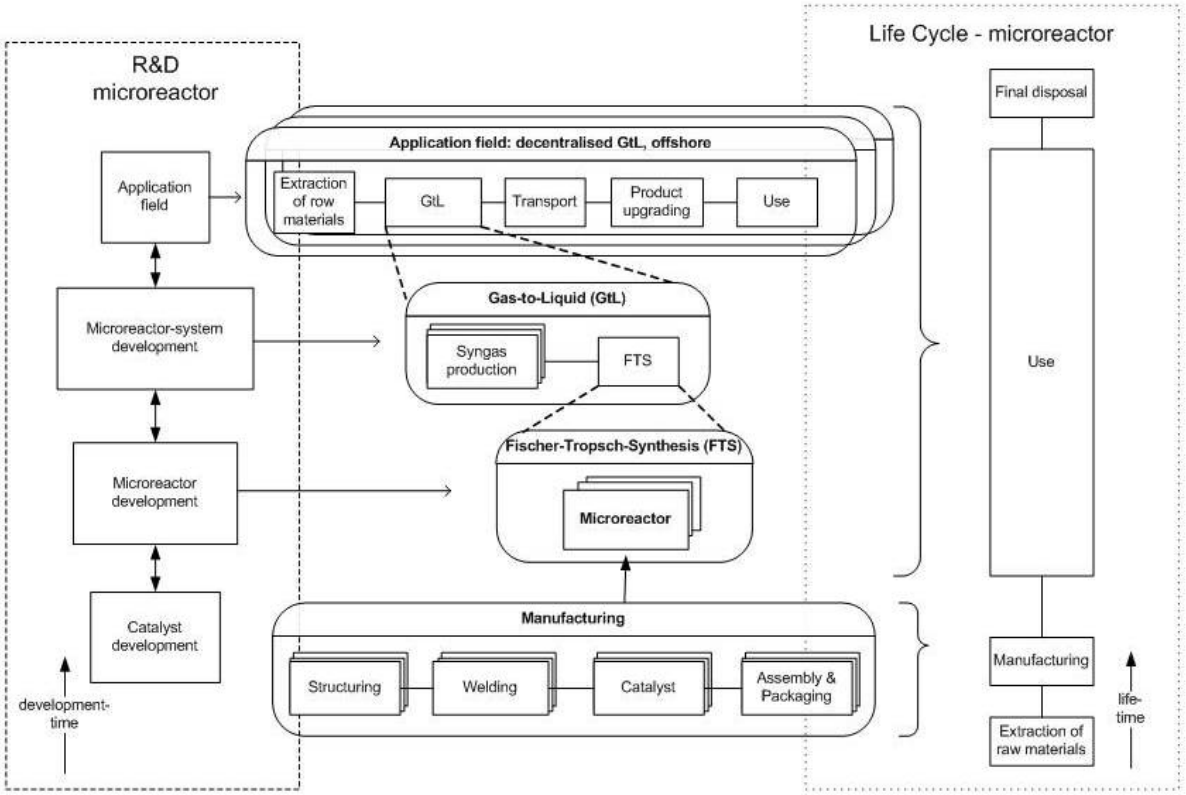

Figure 4.5: Interdependencies between R&D and the life cycle of a microreactor through microreactor design, manufacturing methods, product distribution in FTS and process configuration with GtL application (*Zschieschang et al.*, 2012b).

Based on the above mentioned challenges and the results from Section 2.4 the following requirements towards a new methodology for applying LCA to a microreactor development model can be formulated.

- Incorporating LCI modeling in R&D modeling
- Handling of different data types (material, energy, information)
- Back tracing of LCA results in current R&D phase and process
- Incorporating multiple system boundaries
- Incorporating multiple interests/stakeholders (models)

Therefore, the new methodology should allow recursive, iterative and time-dependent modeling.

4.3 Modular Server-Cient-Server methodology

Based on the considerations mentioned in Section 4.2, the new methodology for using LCA in early stage R&D of technology development combines two main principles. First, reducing complexity by splitting the system in subsystems with smaller system boundaries and second, combining models in technology development and LCA.

The first principle, which will enable the use of LCA according to the goal and scope definition, requires decomposition of a complex system into smaller parts, so called modules, with a single technical component, stakeholder, systems boundary and intended use. This first part is strongly related to the modeling of the microreactor development in Section 4.1. In contrast to the simplified microreactor development model, emphasis is here put on all processes relevant in the life cycle of the technical components within and between the development phases.

The second principle is based on the idea of combining different models to allow back tracing of LCA results to the R&D model, see Section 4.2, Figure 4.4. Both scientific approaches, LCA and technology development (R&D), already use appropriate but differing methods and tools towards the analysis of their respective system. Avoiding time- and cost intensive model and method development, existing tools can be linked from both sides via so called interfaces. An interface is here understood as translator for different data types occurring in both scientific fields. Compared to the iterative approaches in common LCA, advantages of the second principle include easy technical integration in R&D (i.e. no change in software tools) and the possibility for time- and cost saving automation (only one simulation for technical and LCA results). Furthermore, interfaces allow the interconnection between differing R&D models within the technology development. Because of this interconnections, recursive modeling can be implemented. Despite, it is assumed that direct interconnections only appear between R&D models and not between LCA models. LCA models are defined as translated R&D models fulfilling the requirements for LCI modeling and material flow analysis (see Section 2.1.1).

To cope with the different data requirements within R&D and LCA, various sophisticated software tools exist in LCA as well as in R&D. To realize the second main principle of the new methodology, the Client-Server concept from the network architecture in computer science is introduced. This concept allows connecting R&D with LCA as well as of different models within the two.

Thus the new methodology is called MSCS methodology. Within the following section, each part of the MSCS methodology is explained in more detail.

4.3.1 Modularization

Modularization is required in three contexts:

1. R&D and LCA
2. Technology development phases
3. Technical components within one development phase

Consistent with the required contexts, modularization is done on two levels; horizontal and vertical as shown in Figure 4.6. In vertical direction, the modules are again part of a higher level module. According to the general systems theory, this vertical modularization is equivalent to a system hierarchy (see Section 3.1). Therefore, vertical modularization is here used for the modeling of technology development phases and system complexity and thus represent the time factor.

Horizontal modularization serves two functions; first, to illustrate the LCA and R&D model of a considered module and second, to allow modeling of the technical components or processes within one development phase.

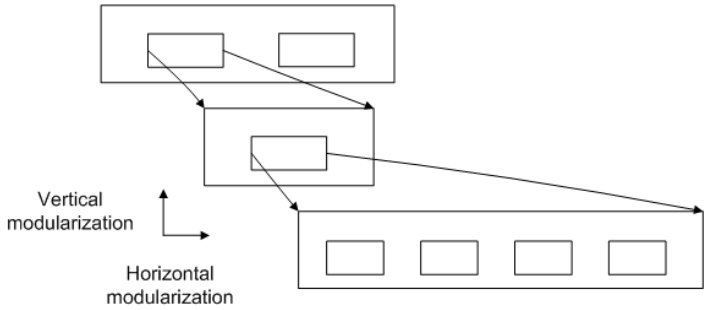

Figure 4.6: Vertical and horizontal modularization of a system.

The structure of a module follows the structure of an object as defined in object-oriented software engineering illustrated in Figure 4.7. Each module is defined by one or more attributes and methods. Client, Server and Interfaces are defined as modules and therefore have identical structures.

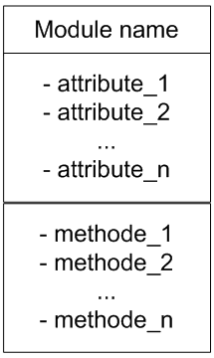

Figure 4.7: Module structure with different attributes and methods. Adapted from the UML notation (*Booch et al.*, 2004).

4.3.2 Server and client

In software engineering the server and the client approach is used for distributed applications, where the server is an application offering a service to the client. In order to guarantee correct data transfer between models in R&D and LCA, a server is used on two levels: 1) on the software level and 2) on the content level as illustrated in Figure 4.8. The focus in this work lies on the content level, nevertheless some aspects towards the software level are explained and illustrated. The client and server approach is used to combine different data types within R&D models and withing LCA models. Both, the R&D and the LCA models provide data, which require translation via so called interfaces. In this context, R&D and LCA models thus act as server and the interface as client.

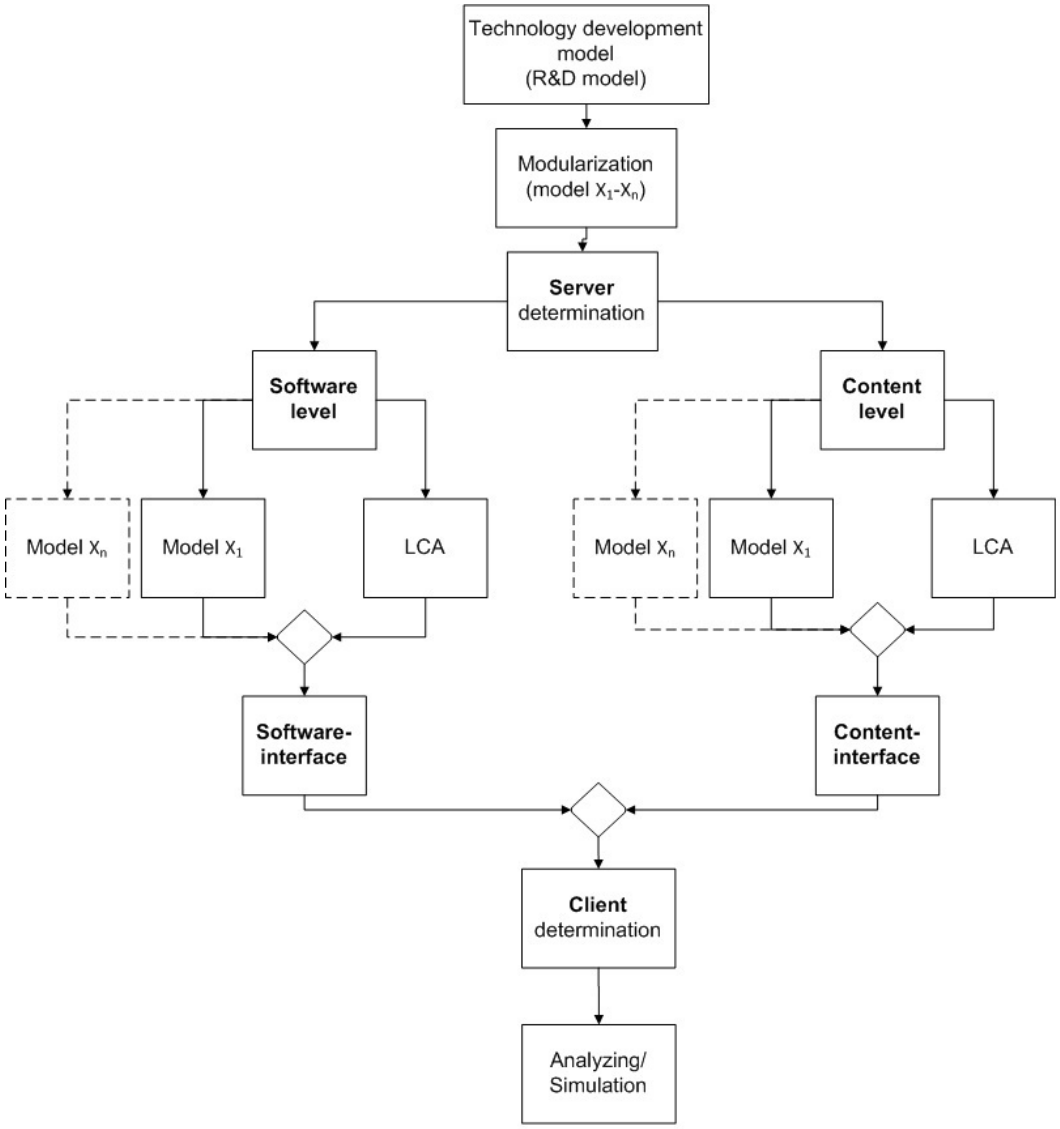

Figure 4.8: Content and software level within the MSCS methodology.

4.3.3 Interfaces

Interfaces are defined as modules similar to the definition of an object in object-oriented software engineering. Therefore, just as a module, an interface consists of attributes and methods. Boundary condition for interface attributes is that each model, connected by the interface, provides methods for this attribute. Similar to the client and the server, interfaces are used on the content and the software levels as shown in Figure 4.8. On content level, content-interfaces are used within different R&D models and between R&D and LCA models. Content-interfaces are supposed to allow back tracing LCA results in the R&D model as well as translating R&D model information to the LCA model via the client.

In order to avoid interface conflicts on software level an appropriate software-interface must be chosen with respect to existing software tools in R&D and LCA. Therefore it is necessary that the client has access to both servers on the software level via the software-interfaces. Common software tools in microprocess engineering R&D are for example Matlab®, Aspen®, CAD, or Comsol® and on LCA side openLCA®, Umberto® or Gabi®. Efficient parametric modeling on one or more computers requires the availability of all three components (two servers, one client) within the same run-time system to use an inter-process communication (IPC) interface. Examples for IPC's are Distributed Component Object Model (DCOM), Common Object Request Broker Architecture (COBRA) or the Hypertext Transfer Protocol (HTTP) as a network protocol.

Finding the right content interfaces is the most important part of the Server - Client - Server concept. The aim of the interface is the functional relation between R&D methods for technology development with LCI analysis methods. In other words, it is necessary to find parameter-based connections between different data types and contents. Parameterized processes on R&D side are inferior for this application.

4.4 Workflow for combined application

First of all, the entire technical development should be analyzed with respect to the life cycle approach, which means to determine the product (technology) and possible application fields. This normally involves increasing technical complexity and prospective thinking due to the fact that possible application fields are not necessarily conceivable at early stages of development especially for single technical components and new developments. Such procedure follows the first step of the sensitivity model by Frederic Vester (*Vester*, 2007) and is also part of the integrated product and process management (*Eversheim*, 2005).

After establishing an overview over the whole system it should be decomposed into modules by using one of the four existing decomposition strategies. These decomposition strategies should be chosen according to the methodologies that would be most appropriate for the intended interest in the analyzed model and might already be part of the specific technology development. This procedure is necessary to get an overview over the used models in R&D in order to identify interdependencies between the models and relevant parameters. Identification of relevant parameters follows in the second part 'Recognition' of the sensitivity model (see Section 3.2), which includes registering involved variables, checking variables for systemic relevance, studying interactions, determining the role within the system and examining overall interconnections.

Since the technology development system consists of multiple modules, the identification of relevant parameters has to be done for each module as well as for the whole systems. Exemplary methodologies appropriate for the identification are matrix and pareto analysis, which reduce the number of elements to the most relevant ones (*Gasafi*, 2006, *Kurz*, 2009, *Ophey*, 2005).

In the third step, module sets are defined. A module set contains the LCA model and the corresponding R&D model of the process to be analyzed. The definition of the module sets includes the assignment of server or client properties on the content level as well as on the software level to these two models, as illustrated in Figure 4.9. The software level is of major relevance to facilitate integration in technical development processes. Besides that, the complexity of the calculation algorithm as well as the amount of information require sophisticated software tools.

Within the fourth step, interfaces on content and software level are determined. This step is again of major relevance, because of the required connection between the two models of the module set, the R&D and the LCA model. The determination of the content interface requires the same work steps as the identification of relevant parameters within and between

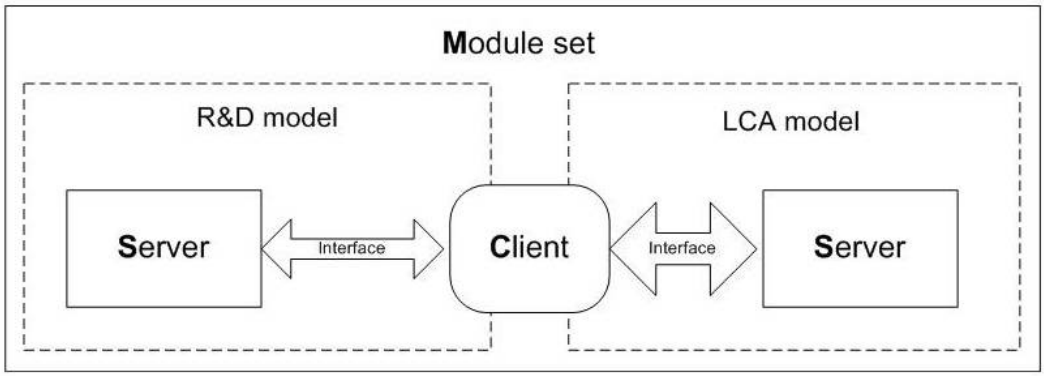

Figure 4.9: Module set according to the MSCS methodology (*Zschieschang et al.*, 2012b).

the R&D modules. Finally, the modules can be analyzed according to the interest.

The four major steps of the MSCS-methodology are summarized below:

1. Modeling based on the general systems analysis
2. Modularization
3. Determination of server and clients on content and software level
4. Determination of interfaces on content and software level

4.5 LCA software development

As mentioned in Section 2.1.2, LCA as a methodology provides a guideline on how to do a life cycle analysis, but for the practical implementation specific software tools are necessary. The software tools for the inventory of the product system in LCI are developed for material flow analysis calculation, in particular for solving linear systems by matrix inversion and Gaussian algorithms. Within the black-box modeling of the considered process, the black-box is described by a functional relation between the in- and output flows, neglecting further process information. However, such further process information is of major relevance in R&D of technologies.

Since the LCI methodology was not developed for the specific requirements in technology development, current software tools need to be developed further to allow the use of the MSCS methodology, and therefore the use of LCA in technology development.

Design processes, an integral part of technology development do not contain any material or energy flows, but design specific information such as technical component geometry, material properties, or process conditions for the chemical reaction. Moreover, such information in turn influences the material and energy flows in manufacturing and use of the considered technical component. In order to get design specif environmental information in manufacturing, use and recycling of the technical component under development, foreground processes in LCI need to be parameterized. Parameterized processes allow necessary material and energy calculations based on design specific information. Thus, the design specific information is related to the process parameters, which are again related to the material and energy flows.

In summary following software implementations are required for the use of the MSCS methodology:

1. Design process as own transition in the flow chart
2. Implementation of information flows in addition to material and energy flows in MFA within LCI
3. Implementation of parameterized foreground processes in manufacturing and use of the microreactor

Since the implementation strongly depends on the utilized LCA software-tool and the operating system for the LCA and R&D models, no detailed knowledge about the implementation on a practical level is provided here. Information about the software used in the case study can be found in the following Chapter 5.

CHAPTER 5

Case Study

In this chapter the case study of the microreactor development for FTS in a GtL process is analyzed by using the novel MSCS methodology for the application of LCA in early-stage technology development. Section 5.1 summarizes the LCA related research question with respect to the case study. A detailed model of the microreactor development is presented in Section 5.2. Based on this model, a LCA is presented for each module in Section 5.3 - 5.6. Finally, a system analysis is undertaken in Section 5.7 using the results from the previous sections.

5.1 Research questions in microreactor development

As mentioned previously, the overall goal of the microreactor development is its application to substitute gas flaring during oil production. Since the microreactor development is a complex system containing the three development phases ***Microreactor***, ***FTS***, ***GtL*** from the simplified microreactor model in Section 4.1 and ***GtL offshore*** as application field, the overall goal has to be specified to the module content. In Table 5.1 LCA specific key questions are answered for each module of the microreactor development.

Module	Interest	System boundary	Functional unit	Stakeholder	Impact assessment indicator	Optimisation
Microreactor	influence of design parameters and manufacturing processes for three microreactor scales	raw material extraction, microreator manufacturing (cradle-to-gate)	1 microreactor with 10, 50, or 100 kg C_{5+} h^{-1}	engineer	CO_2-Eq.	no
FTS	influence of the chain growth probability	use of the microreactor (gate-to-gate)	kg C_{5+} h^{-1}	chemist	CO_2-Eq.	no
GtL	influence of operating conditions	use of the FTS microreactor in the GtL process (gate-to-gate)	kg C_{5+} h^{-1}	process engineer	CO_2-Eq.	no
GtL offshore	most appropriate microreactor scale and GtL design	substitution of gas flaring in oil production	kg CH_4 h^{-1}	investor	- CO_2 reduction - refund in Euro	yes, for given boundary conditions of the oil platform

Table 5.1: Questions related to LCA application within the three microreactor development phases and the application field.

5.2 Detailed microreactor development model

Based on the research questions proposed for the three development phases and the application field of the microreactor in Section 5.1, the microreactor development is subdivided into the following module-sets.

- GtL offshore
- GtL process with
  - FTS
- FTS
- Microreactor with
  - Manufacturing with
    - Structuring by
      - chemical etching
      - milling
    - Bonding (of structured plates)
    - Catalyst
    - Packaging & Sealing with
      - pressure vessel
      - flanges
      - flow distribution structure
  - Design
  - Catalyst development

A formal representation of the above illustrated hierarchical module-sets is presented in Figure 5.1. The time-dependencies are represented by the hierarchy starting with low order (S''') to high order (S^{++}) module-sets. A change in the module attribute might result in a change of the module function and relation of the considered hierarchy. This would results in a change of the system function of the module with a higher hierarchy. In other words, the system function depends on the sub-functions and relations of the containing subsystems. In order to keep the illustration readable, the description of the attributes, functions

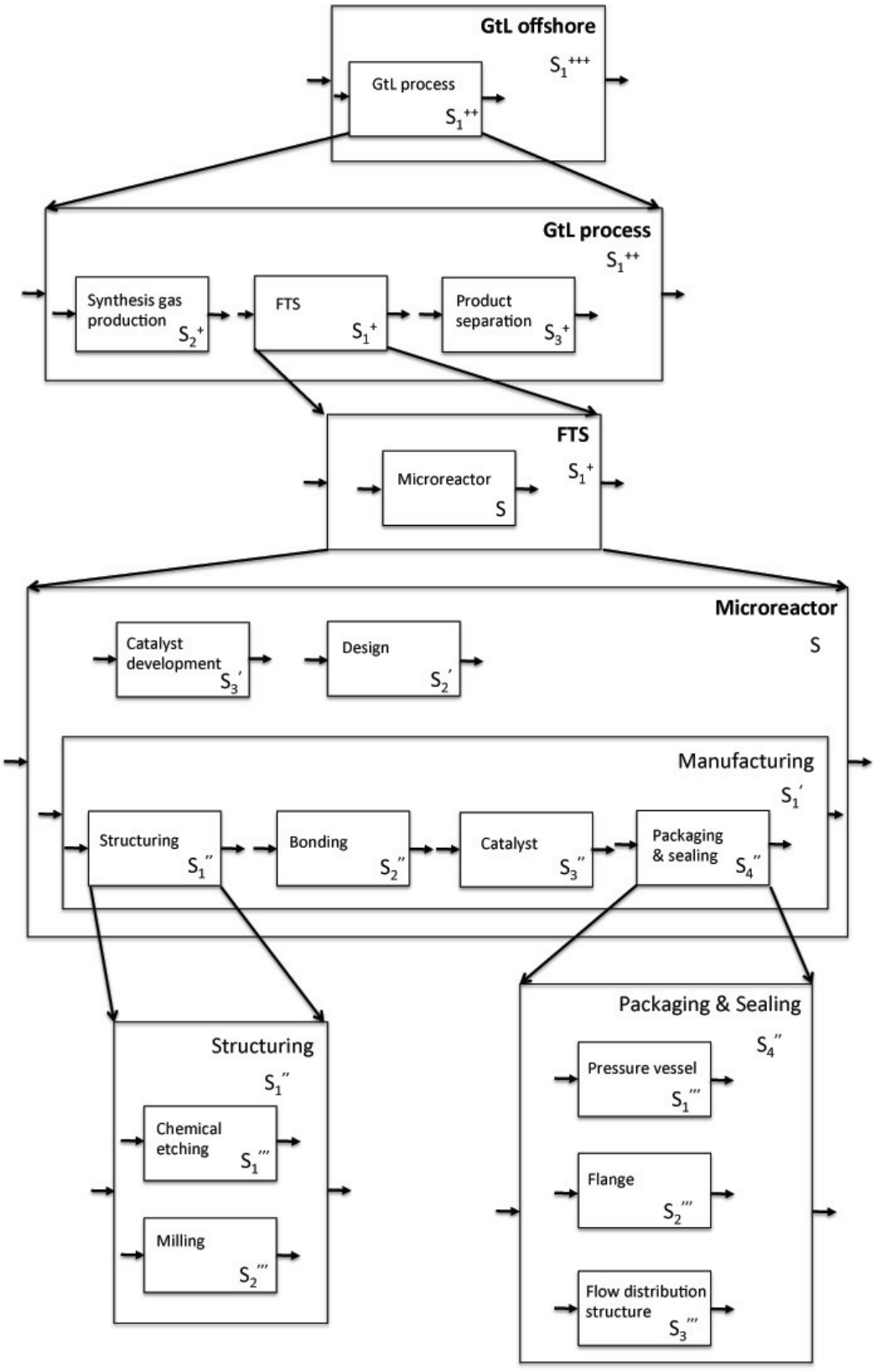

Figure 5.1: Detailed microreactor development model based on the general systems theory definitions.

and relations according to the formal system definition from (*Ropohl*, 1979) is neglected.

The module-sets are modeled according to the specific development phase, purpose and available scientific knowledge. Since modeling the microreactor development involves microreactor design, catalyst development, the simulation of the energy and material consumption of highly sophisticated machine tools for a specific manufacturing task, catalyst preparation, chemical reaction, process simulation and finally application field optimization, the used models are simplified. Based on the resulting models, the module-set is analyzed with the MSCS-methodology starting with a black box LCA model. Subsequently, the interfaces are determined. The final module-set is self-contained to allow the use of different software tools and calculation methods.

In general, a bottom-up design framework is used for the overall microreactor development model. Bottom-up means starting with microreactor design options in manufacturing, checking these designs in FTS and GtL processes and finally determining their appropriateness in offshore application. Calculation methods and software interfaces are set according to this bottom-up model. The developed model is not yet considered for the top-down approach starting with boundary conditions in offshore application.

5.3 Microreactor

5.3.1 Introduction

In order to successfully evaluate the potential of a microreactor for fuel conversion to replace gas flaring, the green house gas emissions of multiple microreactor designs must be assessed in advance to allow selecting the most appropriate design. It is thus necessary to understand the impact of different fabrication methods and design parameters to achieve the most sustainable design. Therefore, LCA analysis results should contribute to answer the following questions:

- Impact of different manufacturing methods for one microreactor design
- Impact comparison between the main manufacturing processes
- Impact of the structuring process by categories
- Impact of the design and modeling parameters
- Impact of the reactor scale

5.3.2 Methodology

According to the modularization shown in Section 5.2, seven simplified R&D and LCA models are established for the manufacturing process; two for structuring (chemical etching, milling), one for bonding, one for catalyst and three for packaging & sealing (pressure vessel, flanges, flow distribution structure). Besides this, a simplified model for the microreactor design and the catalyst development is established.

5.3.2.1 R&D model

Microreactor design

The microreactor is designed for a specific production of FT product in $kg_{C5+}\ h^{-1}$ ($m_{product} t^{-1}$). Therefore, the reaction channel volume $V_{channel}$ is calculated with the catalyst productivity $p_{catalyst}$, catalyst bulk density $\rho_{catalyst}$ and the catalyst void fraction (f_{cvf}) using Equation (5.1). The total channel volume consists of reaction and cooling channels (Equation (5.3)), whereas the cooling channels are calculated with a reduction factor f using Equation (5.2), which represents the possibility to reduce the cooling channels without loosing the heat transfer capabilities of the plate type microreactor. In the example, the cooling channel

volume is considered to be one fifth of the reaction channel volume. The channel volume is also described by the channel height h, limited by the used metal foil thickness, width b and length l, and also limited by the metal foil size (Equation (5.4)). The metal foils considered for manufacturing are 400 x 300 x 0.15 mm^3. Following the MSCS methodology, $V_{channel}$ is called an R&D design parameter.

$$V_{reactionchannel} = \frac{m_{product}}{p_{catalyst}\rho_{catalyst}f_{cvd}} \tag{5.1}$$

$$V_{coolingchannel} = V_{reactionchannel}f \tag{5.2}$$

$$V_{channel} = V_{reactionchannel} + V_{coolingchannel} \tag{5.3}$$

$$V_{channel} = l \cdot b \cdot h \tag{5.4}$$

In order to analyze the impact of different design parameters to the GWP of the microreactor design, as one aim of the analysis proposed in Section 5.3.1, several design and R&D model parameters are analyzed with respect to the relevance and possible impact to the R&D models used.

Following parameters are selected: within the catalyst model, the parameter *productivity* and *catalyst bulk density*; within the microreactor design, the parameter *catalyst void fraction*, which determines the catalyst filling ratio per reaction channel volume; within the structuring model, the parameter *loss cut*, which determines the loss of the raw metal foil plate when cutting to size; within the bonding model, the parameter *welding loss factor*, which determines the efficiency loss of the diffusion welding machine and within the milling model, the parameter *abrasion factor*, which determines the technical layout of the milling machine tool according to the calculated power consumption.

In Table B.1 of Appendix B all microreactor design parameters for modeling designs A to Q are listed. With exception of the catalyst void fraction, all parameters for design A and B are derived from experimental data (*Myrstad et al.*, 2009).

Structuring

Structuring of metal foils in this work is done either by chemical etching or milling. First chemical etching is explained, followed by milling.

- ***Chemical etching***

Averaged measured data for the chemical etching process of one metal foil indexed with '*blank*' are used for the inventory calculation. Therefore, no R&D model for the process system is necessary. The process data (energy, raw materials and chemicals) y are linked directly by a constant x to the LCI parameter n_{blank} (Equation (5.5)). For example, the wet chemical etching of one stainless steel microstructured plate produces 1 kg of waste and requires 0.3 kg of stainless steel.

- ***Milling***

Milling is another fabrication technique within different options of mechanical micromachining. The power consumption and raw material amount of the milling process are calculated by a function of the R&D interface parameter $V_{channel}$ and n_{blank} (Equation 5.6).

According to the required channel volume and size, the necessary cutting force F_C is calculated by the material dependent specific cutting force k_c and the cutting cross section A, a function of the cutting thickness h and cutting width b, as shown in Equation 5.7. This formula is called *Kienzle equation*. Based on experimental results, the specific cutting force k_c can be further expressed as $k_C = k_{c1.1} \cdot h^{1-z}$, with the material dependent values specific cutting force $k_{c1.1}$ and slope value (1-z) (*Paucksch et al.*, 2008). The indicated angle χ, defined as angle between the main cutting edge and the raw workpiece contour, determines the width b and thickness h of the cut together with the feed rate s and the cutting depth d as shown in Equations 5.8 - 5.9.

The final power consumption E is a function of the cutting force F_C, the cutting speed v, time t and the efficiency loss η (Equation (5.10). Furthermore the milling machine was dimensioned with an abrasion factor for the calculated power consumption to compensate the energy consumption of other machine components such as hydraulic and cooling systems (*Braun*, 2012, *Abele*, 2012).

$$y = x \cdot n_{blank} \tag{5.5}$$

$$y = f(n_{blank}, V_{channel}) \tag{5.6}$$

$$F_c = k_c \cdot A \tag{5.7}$$

$$b = \frac{d}{\sin \chi} \tag{5.8}$$

$$h = s \cdot \sin \chi \tag{5.9}$$

$$E = \frac{F_c \cdot v \cdot t}{\eta} \tag{5.10}$$

The meaning of the individual parameters and their values are summarized in Table 5.2.

Diffusion bonding

The energy demand W for diffusion bonding of the microstructured plates to form a microreactor is calculated with the specific heat capacity c_p of the metal foils (stainless steel 316L), the design-dependent reactor mass $m_{microreactor}$ determined from the first fabrication step Microstructuring, the bonding temperature ΔT defined as $T_{Bonding} - T_{Ambient}$ and the factor f for efficiency loss of the oven as shown in Equations (5.11) and (5.12).

$$c_p = \frac{\Delta E}{m_{microreactor} \Delta T} \tag{5.11}$$

$$W = f \Delta E \tag{5.12}$$

Catalyst

The FTS catalyst preparation method wash-coating consists of four process steps, co-precipitation, evaporation, dehydration and calcination. In this work, only co-precipitation is considered due to data availability. The required raw material quantities of the two considered catalysts are calculated by the weight fractions in the catalyst extracted from experimental data (*Myrstad et al.*, 2009). The the raw materials of the catalyst, i.e. cobalt and nickel nitrate hexahydrate are considered as being produced by dissolution in nitric acid with a mass ratio of 1 to 4 (*Holleman*, 2007).

Packaging & sealing

In packaging and sealing the raw material quantities for the gas distribution structure, flanges and an additional safety pressure vessel are considered. The amount of raw material for the distribution structure is identical (based on empirical data) to the raw material amount for the reaction and cooling channels at the chosen microreactor size. The raw material amount for the flanges is considered as constant (2 kg) in certain size ranges - even when changing the outer geometry of the microreactor.

For the calculation of the safety pressure vessel weight, the Long Term Hydrostatic Pressure Resistance Formula also called Barlow's Formula shown in Equation (5.13) was applied, which determines the required thickness of the pressure vessel. Normal stainless steel with a stress limit of 230 Nmm^{-2} and an overpressure of 1 bar was assumed while expanding the small inventory of the microreactor to the large pressure vessel volume in case of microreactor failure.

$$\sigma = \frac{d\Delta p}{2s} \tag{5.13}$$

This pressure vessel design is coupled via the required diameter d to the microreactor design multiplied with a security factor of 1.1.

Table 5.2 summarizes all applied factors in the equations for manufacturing not mentioned previously in the text but essential for the modeling.

	Parameter	Unit	Productivity	Value
Microreactor Design	$\rho_{catalyst}$	kgm^{-3}	bulk density	1000
Structuring	d	mm	cutting depth	0.1
	s	mm	feed rate	0.04*
	(1-z)		slope value	0.75 (Ck 45V)
	$k_{c1.1}$	Nmm^{-2}	specific cutting force	3811 (steel 1.4301)
	χ	°	indicated angle	60
	v	mm min^{-1}	feed rate	1520*
	η		power efficiency	0.8
	lo_{cut}		cutting loss	0.2
	lo_{etch}		etching loss	0.44
Bonding	c_p	J kg^{-1} K^{-1}	specific heat capacity	477 (steel V2A)
	ΔT	K	temperature difference	1000
	f		factor efficiency loss	0.2

Table 5.2: Modeling parameters for microreactor design, structuring, and assembly and bonding. Data source: *(*Eichhorn*, 2011).

5.3.2.2 LCA model

The cradle to gate LCA model for the FTS microreactor manufacturing consists of four parts; Structuring, Bonding, Catalyst and Packaging and Sealing and the microreactor design R&D as illustrated in Figure 5.2. Each of these parts contain foreground and background processes exemplary shown for structuring in Figure 5.2. Here the structuring of reaction and cooling channels are foreground processes whereas the production of raw material, energy as well as the supply materials are background processes. Background processes are specified by in- and output (IO) result processes using generic data from Ecoinvent (*Weidema and Hischier*, 2010). Foreground processes follow the MSCS methodology, and therefore consists of a R&D and LCA model connected via interfaces.

In addition to common LCA models, information flows between the foreground processes are implemented to provide data required within the processes. Further information about the implemented LCA models can be found in Appendix B.

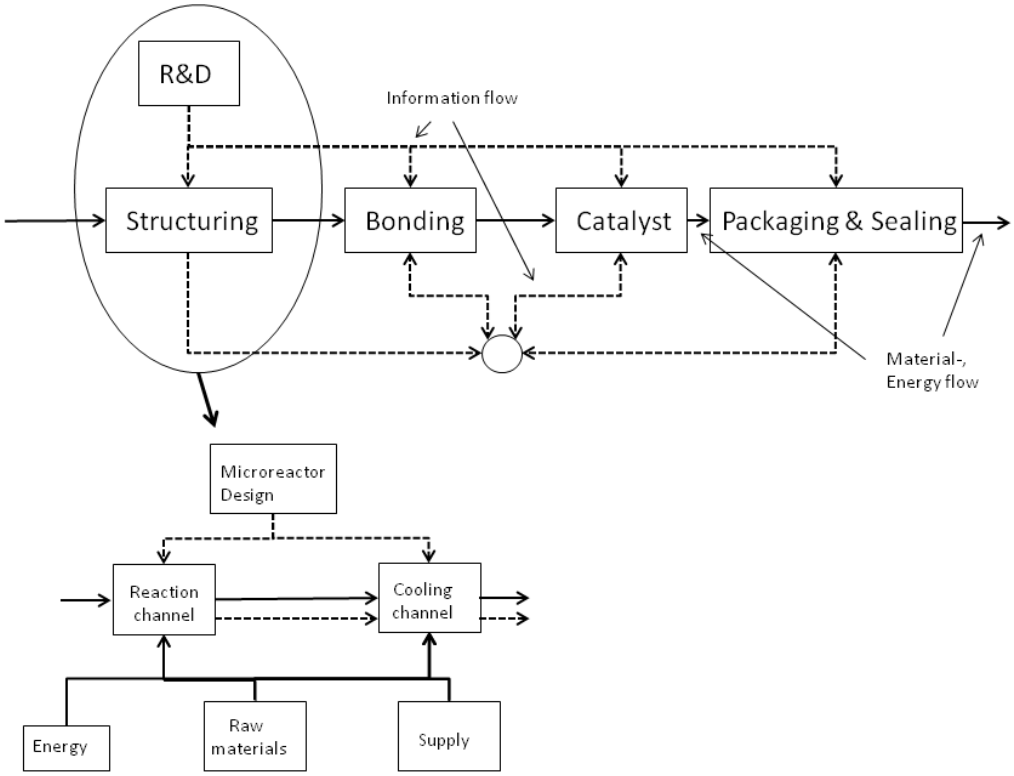

Figure 5.2: Simplified flow-chart of the LCA model for the microreactor fabrication.

5.3.2.3 Interfaces

Within the manufacturing process, interfaces exist in two levels, first between different R&D modules and second between R&D and LCA modules. The attribute of an interface is called parameter.

Equations (5.1) to (5.3) relate the R&D parameter $V_{channel}$, containing reaction and cooling channels, with the catalyst density $\rho_{catalyst}$, the catalyst productivity $p_{catalyst}$, and the estimated output $\dot{m}_{product}$.

Based on the metal plate volume V_{blank} (related to foil geometry) Equation (5.14) calculates the interface parameter $V_{\frac{channel}{blank}}$ with the help of volume specific process parameters such as cutting losses lo_{cut} of the metal foil and etching respectively milling losses $lo_{etch/mill}$. The parameter $lo_{etch/mill}$ is described as loss, because this metal foil volume is used as reactor core and therefore not available for channels. Based on this, the LCI parameter $V_{\frac{channel}{blank}}$ for both structuring processes can be determined from Equation (5.15).

Compared to the wet chemical etching the milling process is based on a parameterized machine model for calculating the power consumption and uses $V_{channel}$ as R&D and interface parameter. The raw material consumption is calculated similar to the wet chemical etching process with $V_{channel}$ as R&D, $V_{\frac{channel}{blank}}$ as interface and $V_{\frac{channel}{blank}}$ as LCI parameter.

$$V_{\frac{channel}{blank}} = V_{blank} lo_{cut} lo_{etch} \tag{5.14}$$

$$n_{blank} = \frac{V_{blank}}{V_{\frac{channel}{blank}}} \tag{5.15}$$

Figure 5.3 illustrates the identified R&D, interface and LCI parameters for the two considered structuring processes wet chemical etching and milling.

The power consumption for bonding is calculated via the specific heat capacity of the microreactor as described in Section 5.3.2.1. Therefore the total *mass of the structured microreactor* is used as interface parameter between the R&D and LCA model calculated by the structured foil mass multiplied with the foil number. Raw material quantities of the catalyst components are calculated by the interface parameter total *catalyst mass* and the *specific weight fraction* of the components.

In addition the *catalyst mass* is calculated by determining the quotient from the output

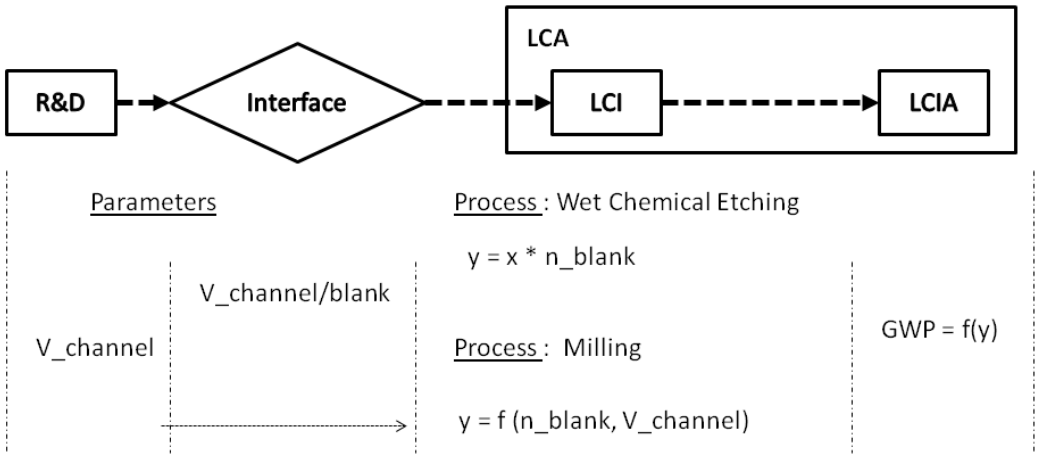

Figure 5.3: R&D, interface and LCI parameters for wet chemical etching and milling.

and the catalyst productivity. Within microreactor packaging & sealing, interface parameters are required only for the pressure vessel since the flange mass is constant and the distribution structure is a multiple of the reaction channel mass. The pressure vessel mass is calculated via the Barlow's Formula and therefore the microreactor diameter is used as interface parameter.

5.3.2.4 Software

The software Umberto®(*Hedemann*, 1998-2012) is utilized for LCA analysis with CML2001 (*Guinee*, 2001a,b) as Life Cycle Impact Assessment method, and Global Warming Potential (GWP in kg CO_2-Equivalent) as impact indicator. Other impact indicators are not considered due to the primary goal, i.e. the reduction of green house gas emissions by the use of microreactors for fuel conversion instead of gas flaring.

Processes are specified by functions and parameters written in JavaScript within Umberto® as shown in Appendix B. The reactor design and fabrication processes as well as the fabrication processes are interconnected by using data flows as illustrated in Figure 5.4. Since Umberto®only provides fixed transition and network parameters not variable by calculation within the transition specification, material flows are implemented as information flows. Ecoinvent v2.2 is used as database for generic data such as electricity and raw materials (*Weidema and Hischier*, 2010). All utilized result processes are named in Appendix B.

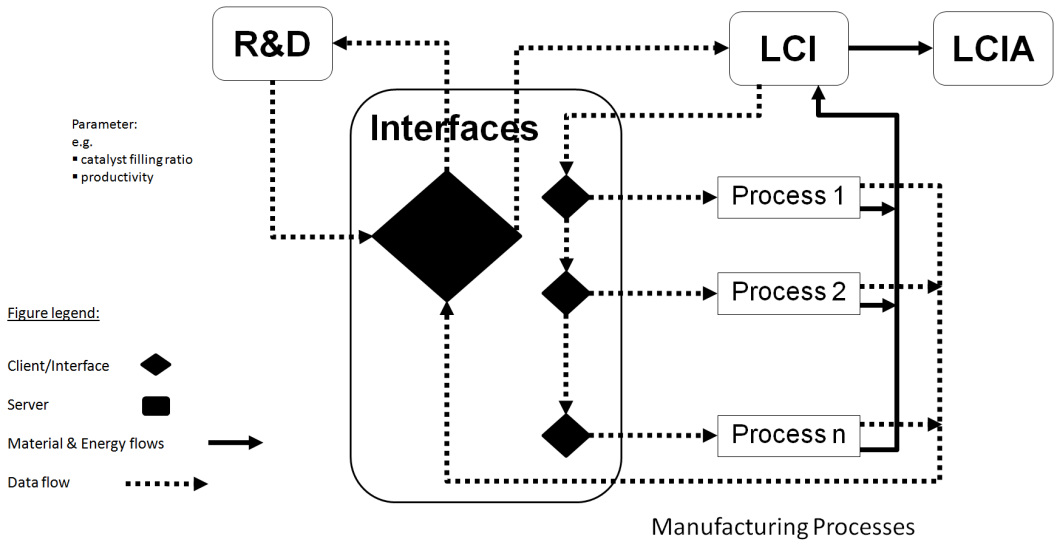

Figure 5.4: Modular Server-Client-Server methodology applied to the example of micro-reactor manufacturing.

5.3.3 Results

5.3.3.1 Impact of different manufacturing methods for one microreactor design

The GWPs for design A (see Table B.1 in Appendix B) are illustrated in Figure 5.5 according to the used structuring process for reaction and cooling channels. In addition to wet chemical etching and mechanical micromachining, the third bar in Figure 5.5 illustrate a combination of both. Wet chemical etching is used for structuring of the reaction channels and micromechanical machining for the cooling channels. Microreactors manufactured using micromechanical machining (milling) show the lowest GWP for all designs.

These results indicate that the structuring fabrication step by wet chemical etching is inferior. A complete fabrication by milling reaction and cooling channels seems advantageous at a discussion level without consideration of GWP savings in the GtL process due to the application of microreactor technology. Life cycle costing (LCC) is not addressed here, but may show different results.

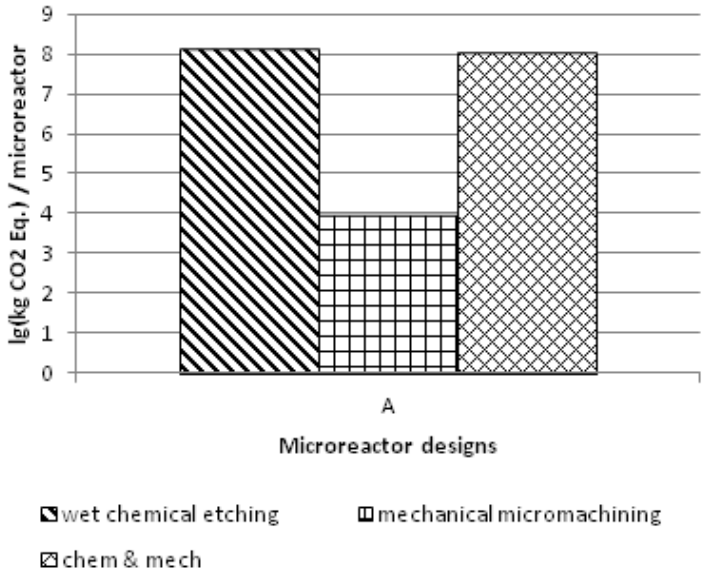

Figure 5.5: Total GWP impact of three different fabrication processes for design A of a microreactor: wet chemical etching, mechanical machining and a combination of both.

5.3.3.2 Impact comparison between the main manufacturing processes

Based on the observed strong influence of the wet chemical etching, existing designs A and B were compared regarding the GWP impact of each fabrication step using wet chemical etching as structuring process. Results are shown in Figure 5.6. As expected the structuring shows the highest impact compared to bonding, catalyst and packaging & sealing.

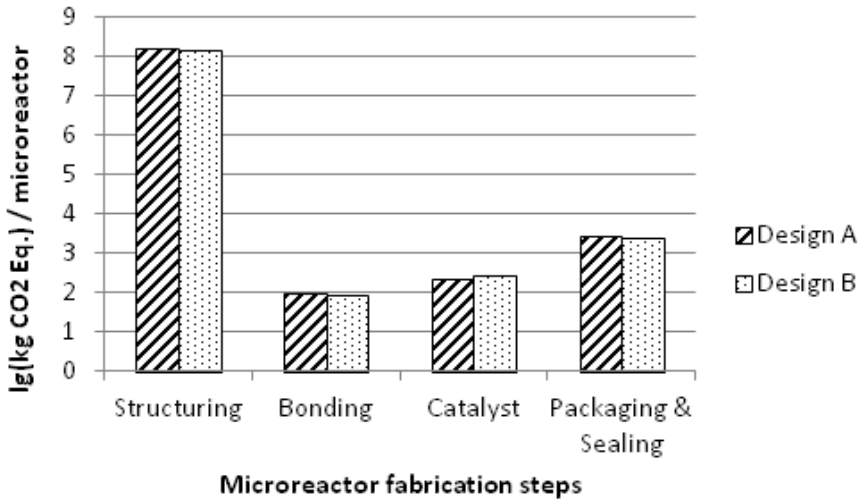

Figure 5.6: Impact of the individual fabrication steps using wet chemical etching as structuring process for microreactor designs A and B on the total GWP of microreactor fabrication.

5.3.3.3 Impact of the structuring process by categories

A detailed analysis of the materials and energy used in the structuring process shows a significant impact of the electricity followed by iron(III)chloride for wet chemical etching illustrated in Figure 5.7. In contrast, stainless steel is the major source of GWP impact for mechanical microstructuring. The level of GWP impact of stainless steel is however similar for wet chemical etching and mechanical microstructuring.

5.3.3.4 Impact of the design and modeling parameters

In addition, further analysis of the influence of R&D parameters were investigated by comparing designs with different parameter settings by varying one parameter at a time.

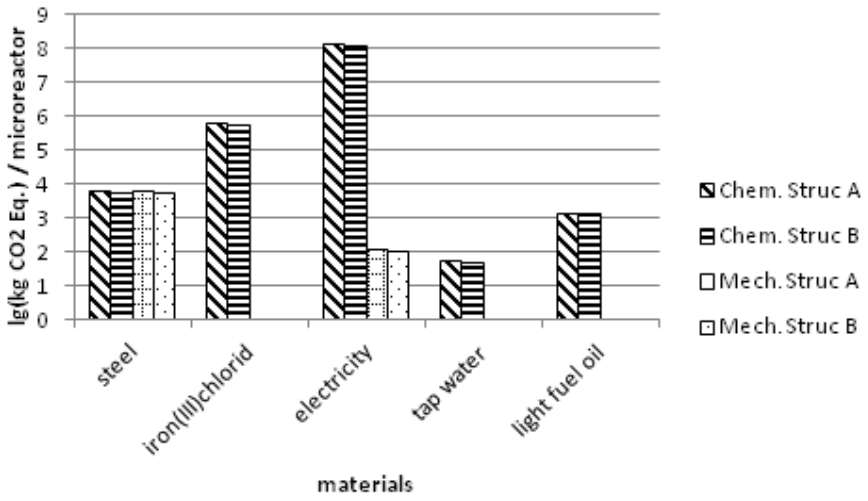

Figure 5.7: GWP Impacts of different sources during microreactor fabrication by wet chemical etching (Chem. Struc) and mechanical microstructuring (Mech. Struc).

Design A, C, D for Co/Ni catalyst and B, G, H for Co catalyst are compared using parameter variations of the catalyst void fraction. Both options with the highest catalyst void fraction of 0.6 show the lowest GWP impact, whereas a Co/Ni catalyst has a higher GWP compared to a Co catalyst as shown in Figure 5.8. Variations of the R&D parameter catalyst productivity show the same tendency; the higher the productivity the lower is the GWP.

To receive information about the relevance of design and process parameters relevant within structuring as crucial microreactor process for GWP impact, a sensitivity analysis was conducted on the example of the Co catalyst. Results for parameters within chemical etching are shown in Figure 5.9 and for milling in Figure 5.10. Obviously, increasing or decreasing the catalyst void fraction shows for both structuring processes the strongest impact on the GWP. Since the catalyst void fraction directly influences the required reaction and cooling channel volume this result was predictable.

5.3.3.5 Impact of the reactor scale

Figure 5.11 shows the GWP for different sizes of a FTS microreactor, i.e. with 10, 50 and 100 kg h^{-1} production rate. Almost no scaling-up effect can be observed due to the strong impact of the structuring.

Due to the decrease of the surface to volume ratio scaling-up reduces the impact per functional unit and thus per product in conventional process engineering. For micro process engineering an internal numbering-up is applied for a scaling-up of the reaction volume.

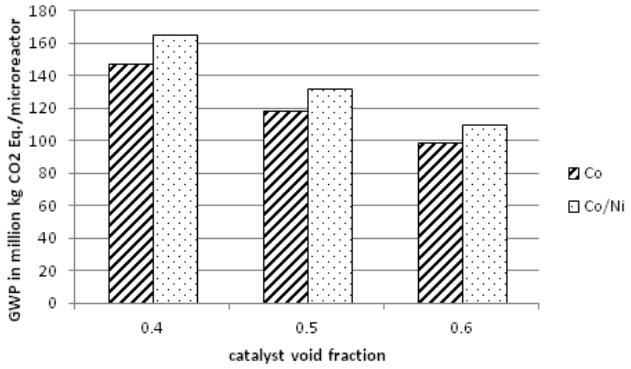

Figure 5.8: Impact of the design parameter catalyst void fraction on the total GWP in microreactor fabrication using two different FTS catalysts.

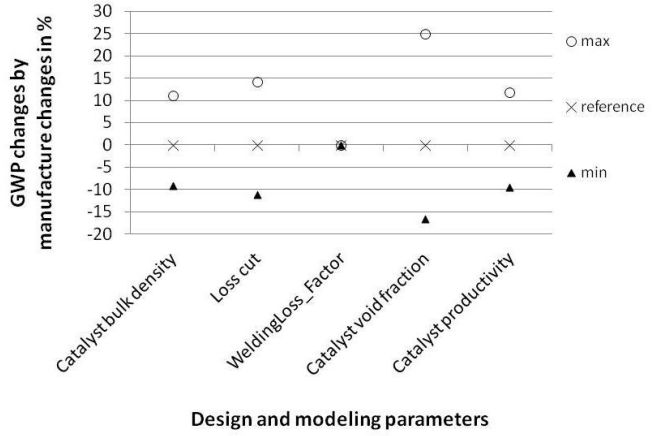

Figure 5.9: Impact of design and process parameters using chemical etching as structuring process.

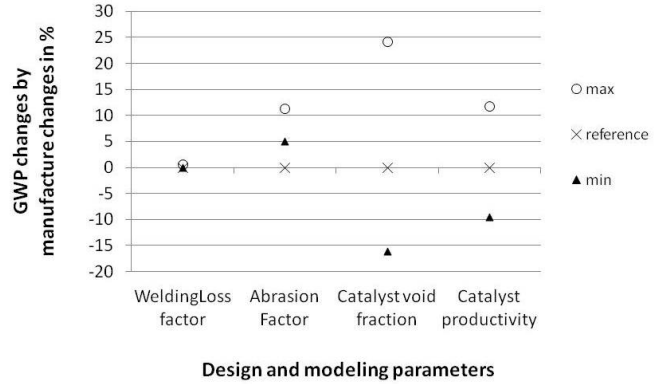

Figure 5.10: Impact of design and process parameters in the milling process.

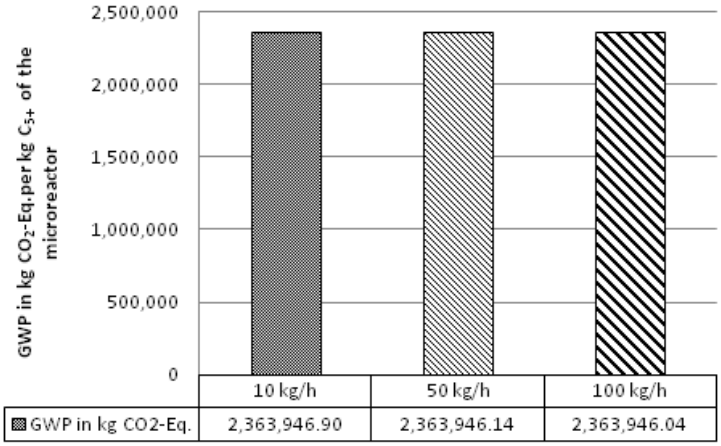

Figure 5.11: GWP for microreactors with different production rates.

Therefore, the surface to volume ratio will not be affected. The amount of raw material and energy per reaction volume stays the same because the reaction volume per microreactor volume is not modified, excluding packaging and sealing. Scaling-up only affects the manufacturing of the components pressure vessel and flange. Because the impact of these two components is largely due to the amount of required steel, which is relatively low compared to the other components as illustrated in Figure 5.12, the overall effect is low.

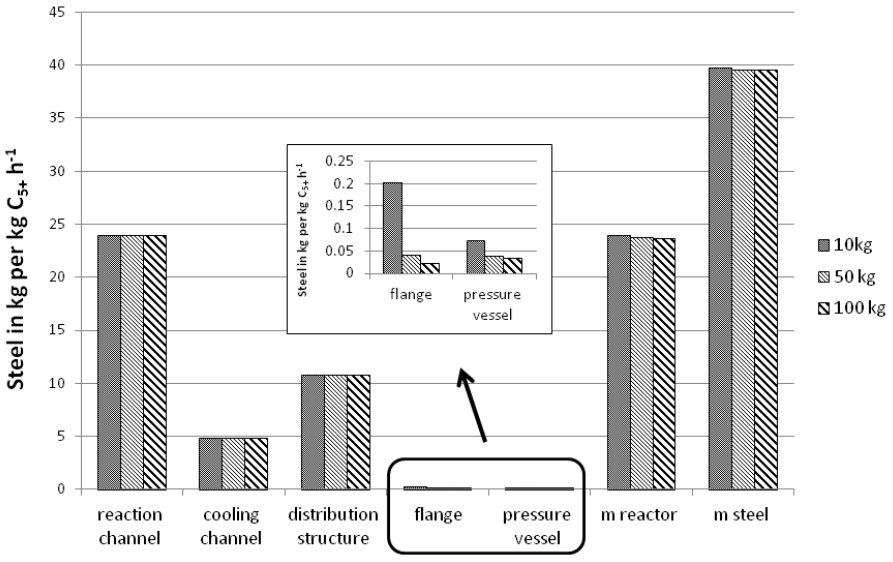

Figure 5.12: Amount of steel used for the microreactor components for three pressure vessel design considerations. The computation of the steel amount is illustrated as JavaScript code in Appendix B.

5.3.4 Discussion

The above mentioned results are based on theoretical models of the manufacturing processes with the exception of chemical etching, which is based on average measured data provided by a company. No model validation with experimental results was possible due to the lack of data, time and resource limitations. The validation of the models is, however, necessary for the comparative use of the received results, especially for the scaling effect.

In micro-mechanical structuring and diffusion bonding, the power consumption is calculated by the reaction volume and by the amount of material linked to the reaction volume. Since this simplified model only represents one aspect within this machining process (neglecting auxiliary systems such as hydraulic circuit, coolant lubricant pump, cooling devices and electric drives) the real power consumption is much higher than the calculated one.

In (*Braun*, 2012), a validated turning operation model indicates that nearly 40 % of the overall consumption results from the cooling and hydraulic aggregates. Even so, the impact of mass production of reaction and cooling channels is not considered in terms of auxiliary systems. Since the results are not considered for decision making but as proof of concept for the MSCS methodology this inaccuracies are justifiable.

Furthermore basic research towards parameterized databases and processes for LCA known as Unit Process Life Cycle Inventory (UPLCI) is still ongoing and software is not available; neither in academia nor commercially (*Kellens et al.*, 2012, *Overcash*, 2012).

5.4 Fischer-Tropsch synthesis

5.4.1 Introduction

The Fischer-Tropsch synthesis is a heterogeneous catalytic reaction and thus strongly influenced by the catalyst properties *lifetime*, *activity* and *selectivity* towards the product. Since there is no convincing evidence to date about the 'correct' mechanism on a molecular level, the prediction of the product distribution remains a challenge (*Steynberg*, 2006). Within all proposed mechanisms the chain growth probability as stepwise procedure is widely accepted (*Steynberg*, 2006). The used Fischer-Tropsch model in Section 5.4.2.1 is based on this assumption and uses the Anderson-Schulz Flory (ASF) distribution for the prediction of the product distribution. In order to include first results of the proof-of-concept reactor (*Myrstad et al.*, 2009) and the desired production rate a modified calculation procedure was implemented.

In order to analyze the impact of a specific microreactor design for FTS on the plant design in the use phase, it is necessary to calculate the reactant and product distribution in advance. Therefore, LCA analysis results should give an answer on the impact according to the chain growth probability.

5.4.2 Methodology

5.4.2.1 R&D model

The Anderson-Schulz Flory distribution in Equation 5.16 is used for the calculation of the product distribution of the FTS. To calculate the product distribution from the experimental C_{5+} selectivity, Equation 5.16 was inserted for carbon number $i = 1$ to 4 in Equation 5.17 and the resulting polynomial function of fifth degree was solved via zero calculation in Matlab®. Furthermore, the calculated chain growth probability (α) was inserted in Equation 5.16 to calculate the individual product selectivities (W_i).

$$W_i = i(1-\alpha)^2\alpha^{i-1} \tag{5.16}$$

$$\sum_{i=1}^{4} W_i = 1 - W_{C_{5+}} \tag{5.17}$$

The prediction of the conversion performance is neglected in this model and assumed

to be 100%. Furthermore, a simplified FTS reaction illustrated in Equation 5.18, with neglected aromatic and aliphatic side products, is used for the reactant and product selectivity calculation according to Equation 5.19 (for reactors without recirculation *Baerns* (2008)). The selectivity $S_{k,i}$ is the quotient between the yield $Y_{k,i}$ and the conversion X_i and equal to the individual product selectivities W_i.

$$\mathrm{nCO} + (2\mathrm{n}+1)\mathrm{H_2} \rightarrow \mathrm{C_nH_{2n+2}} + \mathrm{nH_2O} \tag{5.18}$$

$$S_{k,i} = \frac{Y_{k,i}}{X_i} \tag{5.19}$$

In the following text the workflow for the calculation of the reactant and product flows by given C_{5+} selectivities and production rate as implemented in Matlab®(see Appendix C) is listed:

1. Defining molar weight for C, H, O atoms and H_2, CO, and H_2O
2. Selecting the considered product compositions (alkane, alkene, alcohols) for the calculation
3. Calculating molar weight of the considered products C_1 (methane) to C_4 (butane) and C_{5+}
4. Calculating the chain growth probability α on the basis of C_{5+} selectivity according to Equation 5.16 and 5.17
5. Calculating the product distribution in % wt for each product according to the ASF distribution in Equation 5.16
6. Calculating the weight of the products
7. Calculating the amount of substance of the products
8. Calculating the amount of reactants

Design B from Table B.1 in Appendix B is used as reference with a selectivity W_{C5+} of 84 % and extended by two other design variations with 4 % lower (design B1) and 4 % higher selectivity (design B2) as shown in Table 5.3. All reactors in Table 5.3 are designed for 50 kg h^{-1} C_{5+}.

Design	Catalyst	Temperature T in °C	Productivity in g g^{-1} h^{-1}	Pressure p in bar	selectivity W_{C5+} in %
B1	Co	225	1.9	20	80
B	Co	225	1.9	20	84
B2	Co	225	1.9	20	88

Table 5.3: Variation of design parameters for FTS by changing the C_{5+} selectivity.

5.4.2.2 LCA model

The Fischer-Tropsch model for LCA is a black box model illustrated in Figure (5.13). All in- and output material flows are calculated by the Fischer-Tropsch model in Section 5.4.2.1. This black box model only includes the foreground FTS process, neglecting

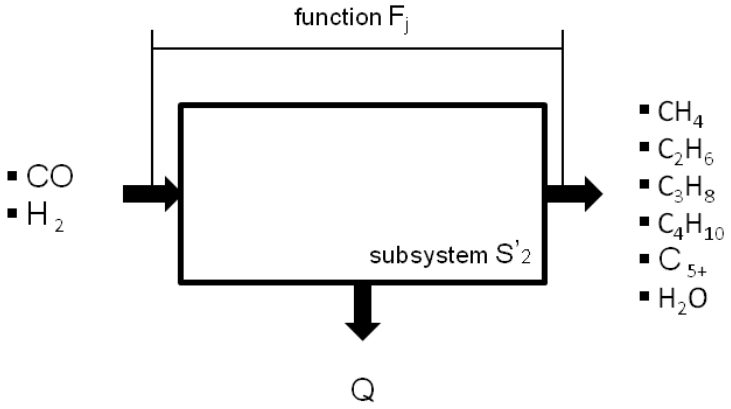

Figure 5.13: LCA model of the FTS process.

upstream and downstream processes such as manufacturing of the microreactor and raw material extraction as well as the cooling systems for the strongly exothermic reaction. Reason for this is that the model only present the FTS process itself and not the life cycle of the reactor or product. The FTS model is part of a module set which is in addition part of a system with higher hierarchy and the assembly is shown later in Section 5.5.

5.4.2.3 Interfaces

Because the R&D model directly calculates the material flows needed within the LCA model there is no need for additional interfaces between the R&D and LCA model. The C_{5+} selectivity is coupled with specific design and operating parameters and therefore used as interface parameter among the vertical modules *Manufacturing* and *GtL*.

5.4.2.4 Software

The chain growth probability α, the reactant and product distribution as well as the material flows are calculated in Matlab® (*Mathworks*, 2012) and Excel based on the desired production rate in $kg_{C5+}\ h^{-1}$ defined within the microreactor design in Section 5.3.2.1. Subsequently, results are transfered to the black box model in the LCA software Umberto®. GWP100a from CML2001 (*Guinee*, 2001a,b) is used as impact indicator within LCIA. The source code for Matlab®is presented in Appendix C.

5.4.3 Results

5.4.3.1 Chain growth probability for experimental data

Using experimental C_{5+} selectivities from (*Myrstad et al.*, 2009) and neglecting reaction and catalyst properties while changing the C_{5+} selectivity arbitrarily, the chain growth probability α increases with higher C_{5+} selectivities as shown in Figure 5.22.

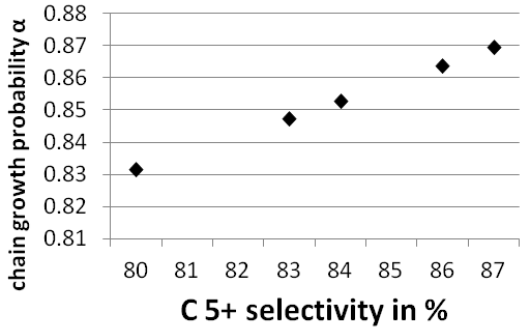

Figure 5.14: Calculated chain growth probability α based on selectivities from (*Myrstad et al.*, 2009) by arbitrary variation of the C_{5+} selectivity.

5.4.3.2 Evaluating experimental results in terms of functional dependency of the chain growth probability on pressure and temperature

Figure 5.15 shows the chain growth probability in terms of temperature and pressure dependency for Design B, with a constant gas hour space velocity (GHSV) of 16200 Nml g_{cat}^{-1} h^{-1} (*Myrstad et al.*, 2009). While increasing temperature at high pressure leads to higher chain growth probability it remains the same at lower pressure. Increasing pressure at high temperature results in higher chain growth probability whereas it remains the same at lower temperature. Due to the poor data availability, the results are good for a first impression but not sufficient for a trend integration in the modeling.

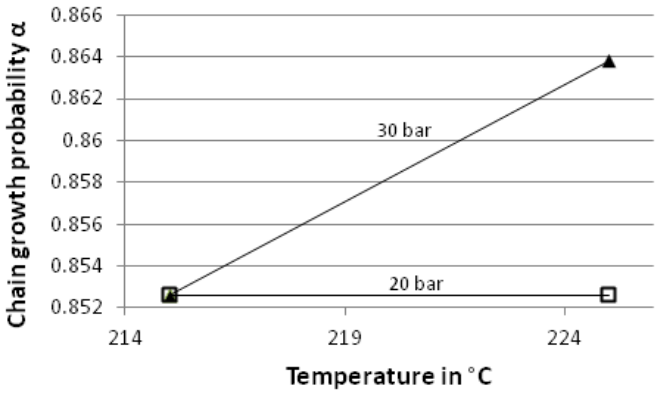

Figure 5.15: Chain growth probability as function of temperature and pressure (exp. data from *Myrstad et al.* (2009)).

5.4.3.3 Comparing these results with other research results

Existing studies towards FTS in a microreactor have not yet examined the chain growth probability as a function of the used catalyst and operating conditions (i.e. temperature and pressure) (*Deshmukh*, 2010), and therefore no validation with other microreactor results is feasible.

Thermodynamics indicate that the generation of small hydrocarbon products becomes more feasible with increased temperatures (*Anderson*, 1952, *Steynberg*, 2006). Consequently, the selectivity of methane should increase whereas the selectivity of C_{5+} should decrease with higher temperatures. Results in Figure 5.15 are not consistent with this assumption however.

Upon increasing the operating pressure for cobalt catalyst the selectivity for higher hydrocarbons (wax) and therefore the chain growth probability increases (*Steynberg*, 2006). The results for FTS in a microreactor in Figure 5.15 are consistent with this assumption for higher temperature.

Since the FTS mechanism on a molecular level is part of ongoing research the transfer of conventional reactor to micro-scale reactor results may not be feasible and needs validation. This validation is beyond the scope of this work.

5.4.3.4 Reactant and product distribution for different C_{5+} selectivities

The reactant and product distribution for different C_{5+} selectivities with the corresponding chain growth probability α is illustrated in Table 5.4. Design B2 with the highest selectivity of 88 % shows the lowest reactant and side product appearance for 50 kg h^{-1} desired C_{5+} product. Since the GWP is only affected by methane as side product, those designs with the highest selectivity may have the lowest GWP.

Design		**B**	**B1**	**B2**
Design parameters	selectivity in %	84	80	88
	α	0.85	0.83	0.88
Reactants	CO in kg h^{-1}	117.2	122.8	112.1
	H_2 in kg h^{-1}	17.7	18.7	16.8
Products	CH_4 in kg h^{-1}	1.3	1.8	0.9
	C_2H_6 in kg h^{-1}	2.2	2.9	1.5
	C_3H_8 in kg h^{-1}	2.8	3.7	2.0
	C_4H_{10} in kg h^{-1}	3.2	4.1	2.4
	C_{5+} in kg h^{-1}	50	50	50
	H_2O in kg h^{-1}	75.4	79.0	72.1

Table 5.4: Reactant and products for different selectivities and chain growth probabilities.

5.4.3.5 GWP impact in relation to the selectivity

The GWP for different reactor designs with different selectivities is shown in Figure 5.16. While providing the same amount of C_{5+} as product, the GWP of the FTS process declines with a higher selectivity and with increasing chain growth probability α, respectively. The reactor characteristic selectivity has no influence within the manufacturing module and thus no influence on the manufacturing GWP.

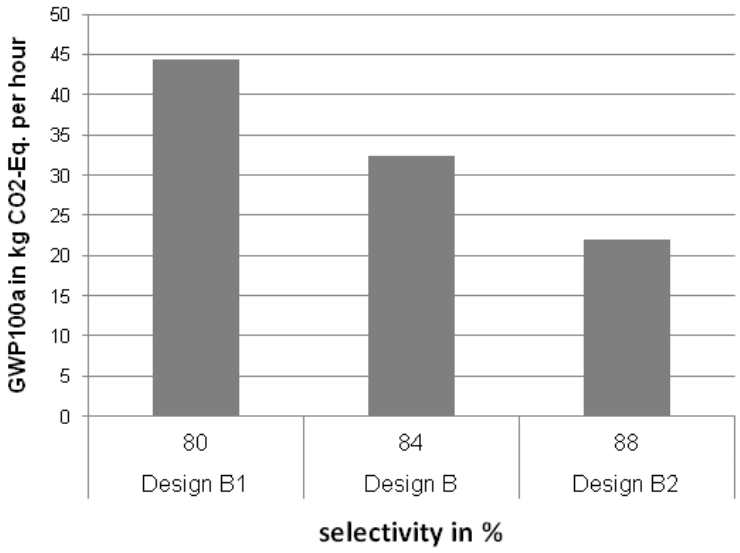

Figure 5.16: GWP of the FTS process for different reactor designs with different selectivities.

5.4.4 Discussion

There are two main uncertainties in the FTS module, one theoretical and one practical. Since research towards FTS kinetics and FTS in microreactors is still ongoing the main area of uncertainty lies within the theoretical R&D model itself. The simplified FTS kinetic model assuming an independent chain growth probability from the chain length is not yet validated for FTS in microreactors. Further uncertainties in this simplification result from neglecting aromatic and aliphatic side products. On top of the theoretical lack of knowledge no practical validation can be carried out.

5.5 Gas-to-Liquid process

5.5.1 Introduction

Within this module an autothermal GtL process converting associated gas to hydrocarbons with the Fischer-Tropsch synthesis was examined in Aspen®Plus (*AspenTech*, 1999). The GtL process is optimized to produce liquid hydrocarbons of the diesel fraction with five or more carbon atoms. Focus is on the conversion of associated gas to syngas by steam reforming and the conversion of syngas to hydrocarbons by FTS. Therefore, the gas purification is neglected in this work. The results of the material and energy balance flowsheets of the GtL process used for the LCA should contribute towards:

- Impact of different process operating configuration
- Impact of different GtL plant scales

For this reason the material flows of the GtL process and resulting GWP are examined towards above mentioned interests.

5.5.2 Methodology

5.5.2.1 R&D model

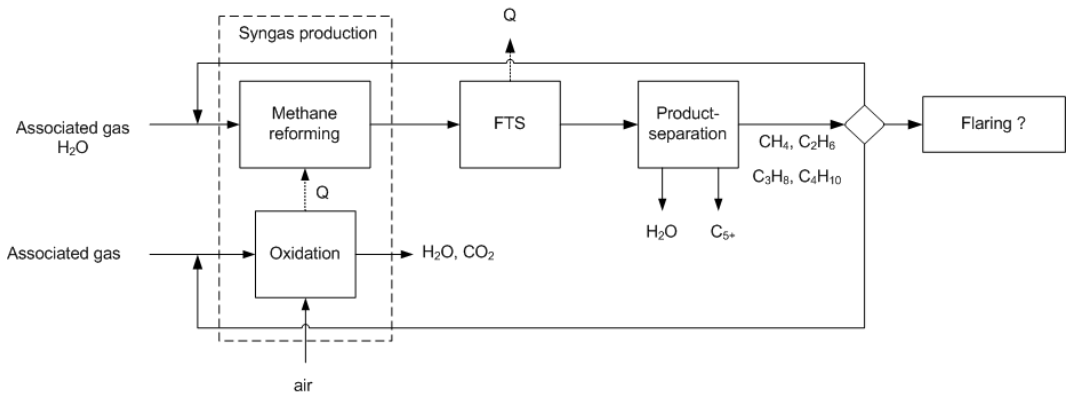

Figure 5.17: Simplified process flow diagram of a GtL process.

The simplified process flow of the GtL system is illustrated in Figure 5.17. It consists of three major processes; syngas production, FTS, and product separation. First, associated gas and water are converted via steam reforming into carbon monoxide and hydrogen. Due to the strong endothermic steam reforming process required heat is produced by methane oxidation. In the second process, the FTS, carbon monoxide and hydrogen are converted

to longer-chain hydrocarbons such as paraffins, olefins and alcohols. Finally hydrocarbons with less than five carbon atoms (gaseous) are separated for recycling to the system whereas hydrocarbons with more than four carbon atoms (liquid) are collected for further use. Two R&D models are required: syngas production via steam reforming and FTS.

Syngas production

The key reaction for the conversion of natural gas into carbon monoxide and hydrogen via steam reforming is illustrated in Equation 5.20 (*Baerns*, 2008). Due to the strong endothermic reaction, the catalytic reaction is carried out at 800-950 °C and 10-40 bar (*Baerns*, 2008). In order to minimize/avoid carbonization as undesirable side reactions, a higher steam-to-carbon ratio is required (*Baerns*, 2008, *Rennard et al.*, 2010). Water gas shift (WGS) as parallel reaction is used to adjust the desired H_2:CO ratio for the FTS (Equation 5.21).

$$C_nH_{2n+2} + nH_2O \rightleftharpoons nCO + (2n+1)H_2 \tag{5.20}$$

$$CO + H_2O \rightleftharpoons CO_2 + H_2 \tag{5.21}$$

FTS

Compared to the FTS model proposed in Section 5.4.2.1, the current FTS model utilizes calculated or given chain growth probabilities for the reactant and product calculation with the ASF distribution by Equation 5.16 in Section 5.4.2.1. The considered workflow in Section 5.4.2.1 is simplified by excluding step 4.

Process parameter settings

Lean associated gas with its gas specifications shown in Table 5.5 is used as feed gas for the syngas production. The operating temperature for steam reforming is set to 900 °C (*Baerns*, 2008), for FTS to 225 °C (*Myrstad et al.*, 2009) and the operating pressure is the

same for both processes. Based on the results from Section 5.4.3.2 the operating parameter pressure is very likely to have an impact on the chain growth probability and therefore to the reactant and product distribution. For this reason and because there are no validated causalities between pressure variation and chain growth probability in microreactors, 24 scenarios with pressure, chain growth probability and Conversion Rate (CR) variations selected according to the FTS results in Section 5.4.3 are examined, see Table D.1 in Appendix D. The CR defines how much of a chemical compound is converted within the reaction time.

Detailed process settings and further parameters for additional components used within the GtL process with the considered R&D models implemented in Aspen®Plus can be found in Appendix D.

feed gas components	amount
N_2 in vol%	0.83
CO_2 in vol%	1.61
CH_4 in vol%	89.64
C_2H_6 in vol%	7.27
C_{3+} in vol%	0.65
Max. total S, vol ppm	4
Hydrogen sulfide, vol ppm (typical)	3
COS, vol ppm (typical)	n.a.
Mercaptans, vol ppm (typical)	1

Table 5.5: Associated lean gas composition (*Steynberg*, 2006).

5.5.2.2 LCA model

The black box LCA model, neglecting up- and downstream processes for material extraction, manufacturing, and the use of the produced product, is illustrated in Figure 5.18. This LCA model considers the in- and output material flows of the whole GtL system but no recycle steams i.e. between syngas production and FTS. Recycle steams are only handled in the Aspen®software

5.5.2.3 Interfaces

The in- and output flows of the GtL system are equal to the IO flows of the black box LCA model as shown in Figure 5.18 and therefore no interface between the R&D and LCA model is necessary.

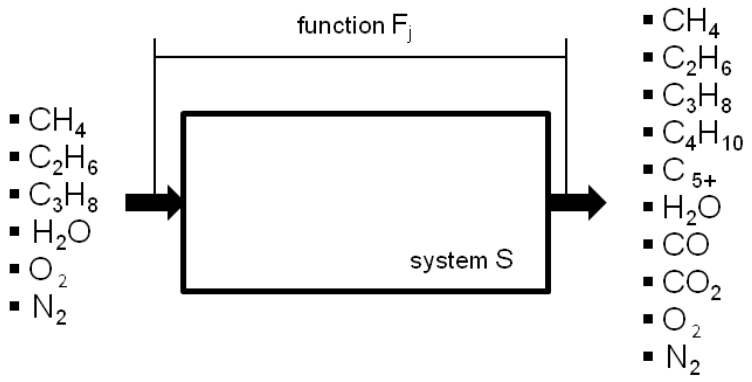

Figure 5.18: LCA black box model of the GtL process.

5.5.2.4 Software

The process flow analysis of the GtL process was done in Aspen®Plus, and LCA in Umberto®. Selected material flows calculated in Aspen®Plus are transfered via VBA script to Excel and with the help of open database connectivity (ODBC) to the black box model in Umberto®. With regard to the MSCS methodology, Aspen®Plus and Umberto® are used as servers, and Excel as client.

5.5.3 Results

5.5.3.1 Material flow analysis of the GtL process

Figure 5.19 and 5.20 show the reactant and product flow results for three GtL scenarios using 20 bar as operating pressure for steam reforming and FTS by varying chain growth probabilities (α) and conversion rate (CR). Scenario 1 and 2 represent real experimental data from (*Myrstad et al.*, 2009) and scenario 12 the best performance with chain growth probability α = 0.9 and CR = 0.9 at same operating pressure.

Having the same throughput per hour of 10 kg h^{-1}, the reactant and product amount is reduced significantly within the range of 9 to 70 % for reactants (see Table 5.8) and 10 - 104 % for products (see Table 5.8) in scenario 12 compared to scenario 1. These results have two effects: first, on the GWP in the FTS process and second by the GtL process design on the GWP in manufacturing. Compared to the GtL process in scenario 12, scenario 1 produced 70% more CO_2 and 30 % more associated gas which is not recovered. Since these two gases are considered in the GWP calculation, the GWP is expected to be around 70 % lower for scenario 12 compared to scenario 1. The second effect concerns the overall feed flow into the GtL process. Increasing the feed flow of the process by constant

desired C_{5+} product output requires a larger size of the reactor. Accordingly, the impact in manufacturing of this equipment will increase due to the larger material and energy amount needed. The compared reactants and products composition for scenario 1 with the 10 kg h^{-1} reactor type are shown in Figure 5.21 and Table 5.8.

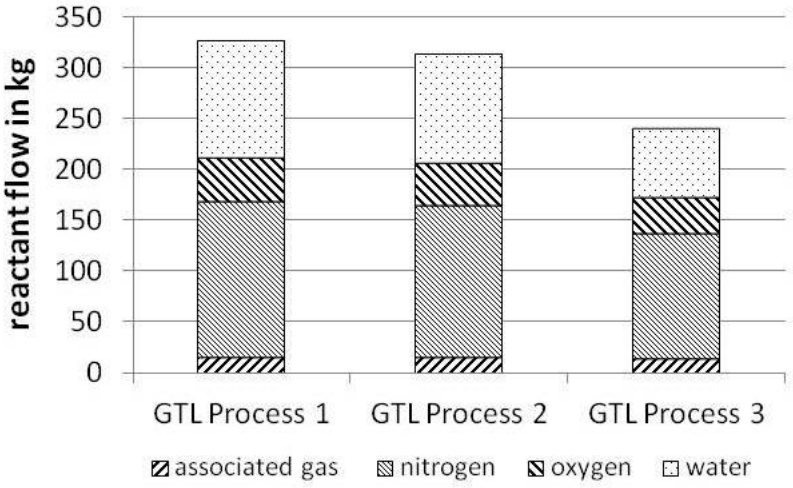

Figure 5.19: Reactant material flow for scenario 1,2 and 12 for the 10 kg h^{-1} reactor type.

Scenario	associated gas in kg h^{-1}	N_2 in kg h^{-1}	O_2 in kg h^{-1}	H_2O in kg h^{-1}
1	14.26	153.17	43.74	116.06
2	13.89	149.76	42.77	107.49
12	13.05	124.08	35.43	68.15
1→12 in %	9	23	23	70

Table 5.6: Reactants of the simulated GtL process in Figure 5.19.

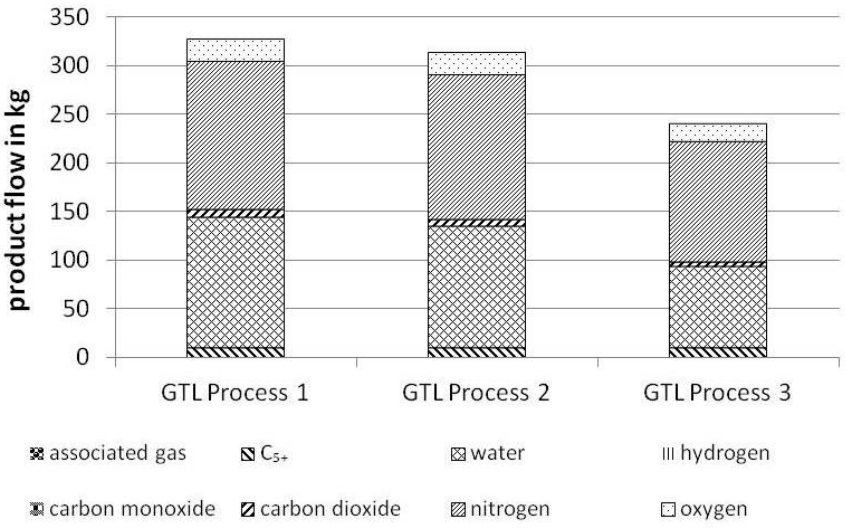

Figure 5.20: Product material flow for scenario 1,2 and 12 for the 10 kg h^{-1} reactor type.

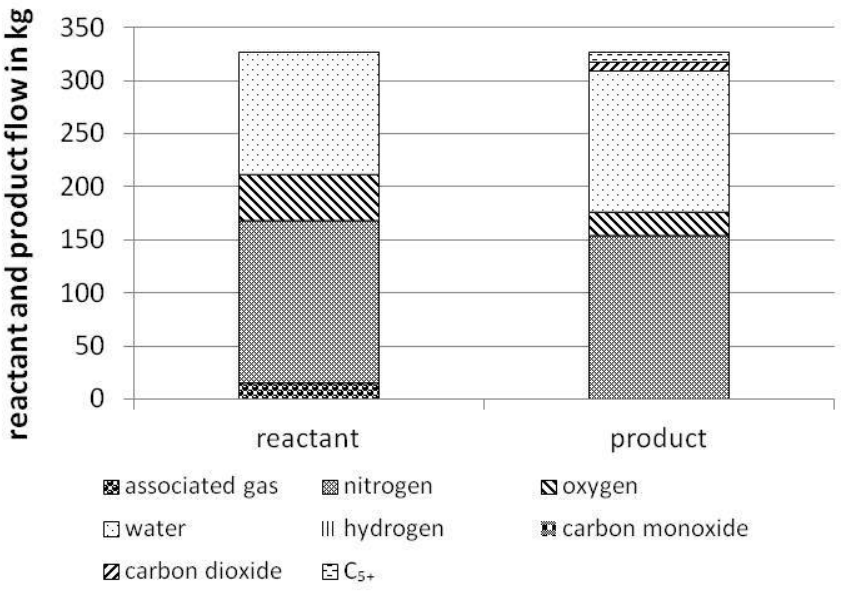

Figure 5.21: Reactant and product composition of scenario 1 for the 10 kg h^{-1} reactor type.

Scenario	associated gas in kg h^{-1}	C_{5+} in kg h^{-1}	H_2O in kg h^{-1}	H_2 in kg h^{-1}	CO in kg h^{-1}	CO_2 in kg h^{-1}	N_2 in kg h^{-1}	O_2 in kg h^{-1}
1	0.11	10	133.45	0.0018	0.007	8.06	153.2	22.4
2	0.08	10	124.15	0.0017	0.007	7.09	149.8	22.8
12	0.08	10	83.04	0.0016	0.003	4.74	124.1	18.8
1→12 in %	30	0	61	10	104	70	23	20

Table 5.7: Product of the simulated GtL process in Figure 5.20.

	associated gas in kg h^{-1}	N_2 in kg h^{-1}	H_2O in kg h^{-1}	O_2 in kg h^{-1}	CO in kg h^{-1}	CO_2 in kg h^{-1}	H_2 in kg h^{-1}	C_{5+} in kg h^{-1}
Reactant	14.26	153.17	116.06	43.74	-	-	-	-
Product	0.11	153.17	133.45	22.4	0.007	8.06	00018	10

Table 5.8: Reactant and Product data of scenario 1 for the simulated GtL process with 10 kg h^{-1} in Figure 5.21.

5.5.3.2 GWP impact of GtL process parameters

- **Chain growth probability**

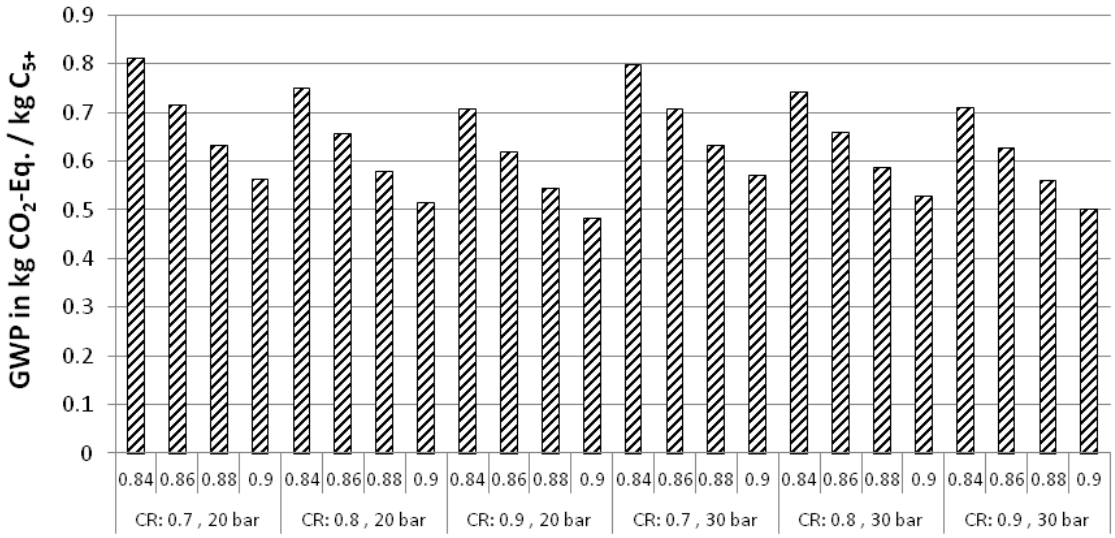

Figure 5.22: Normalised GWP results for scenario 1 - 24 sorted according to chain growth probability variations for the GtL process.

In Figure 5.22 results for scenario 1 - 24 are sorted according to increasing chain growth probability for different parameter sets. Apparently, the GWP depends on the chain growth probability, since in all scenarios having the same CR and pressure, the GWP declines with increasing chain growth probability. The chain growth probability determines the product distribution and therefore the product amount for desired C_{5+}, which is included in the GWP calculation. Scenario 12 with α = 0.9, CR = 0.9, p = 20 bar and GWP = 0.48 kg CO_2-Eq. kg^{-1}_{C5+} h^{-1} has a 69 % lower GWP compared to scenario 1 with 0.81 kg CO_2-Eq. kg^{-1}_{C5+} h^{-1}.

- **Conversion rate**

Increasing the CR results in a lower GWP by constant operating parameters as illustrated in Figure 5.23.

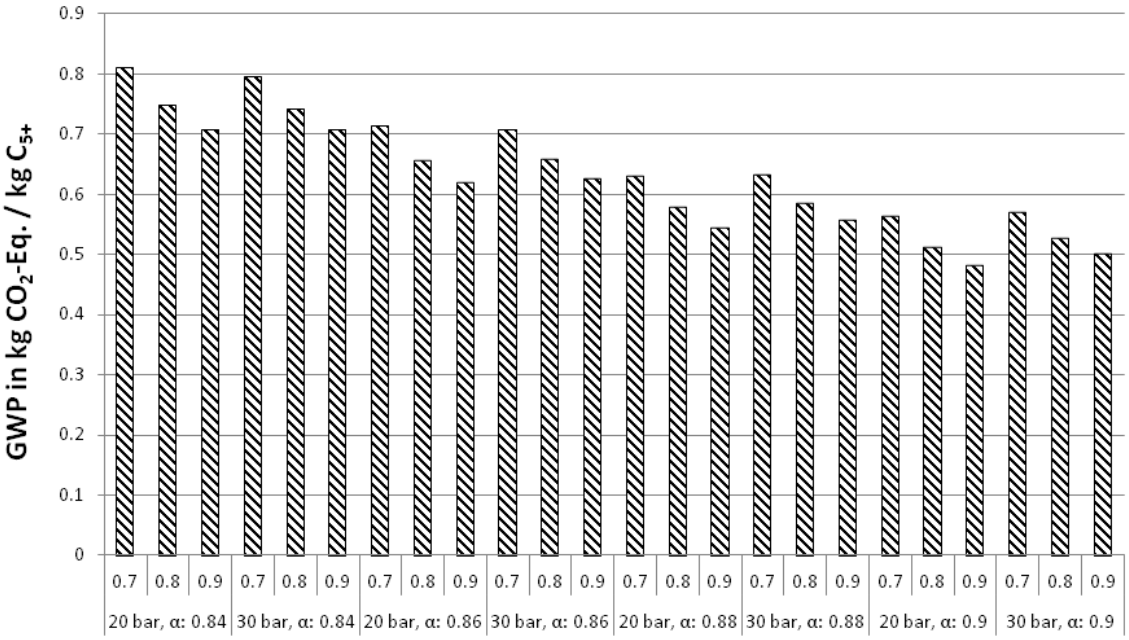

Figure 5.23: Normalized GWP results for scenario 1 - 24 sorted according to CR variations for the GtL process.

- **Pressure**

Results for the pressure influence on GWP shown in Figure 5.24 are less monotonous than for CR and chain growth probability. Scenarios with α = 0.84 and CR = 0.7 and 0.8 as well as with α = 0.86 and CR = 0.7 show a lower GWP for 30 bar as operating pressure than for 20 bar, whereas all other scenarios show the opposite behavior. This can be explained by the influence of CR and α to the feed flows included in GWP calculation, which compensate at α = 0.88 and CR = 0.7, α = 0.86 and CR = 0.8 and α = 0.84 and CR = 0.9.

5.5.3.3 GWP impact of the reactor scale

The normalized GWP impact of scenario 1 - 24 sorted by increasing chain growth probability are shown for three reactor sizes in Figure 5.25. Obviously, there exists only a marginal difference of the GWP impact in kg CO_2-Eq. per kg C_{5+} h^{-1} within this reactor size range.

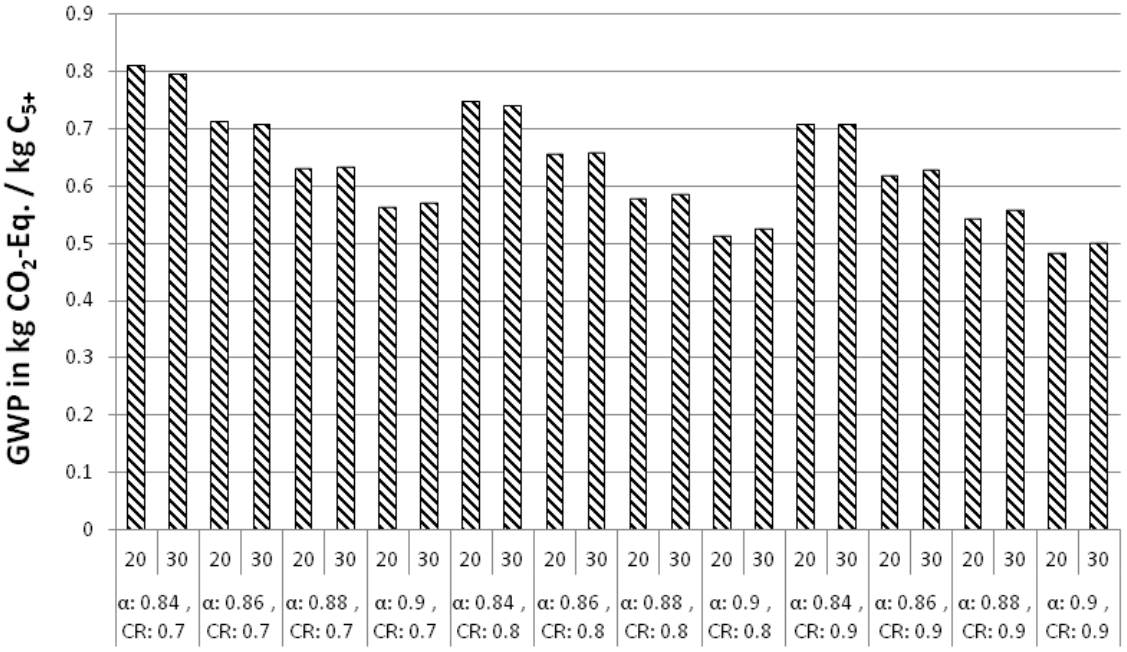

Figure 5.24: Normalized GWP results for scenario 1 - 24 sorted according to pressure variations for the GtL process.

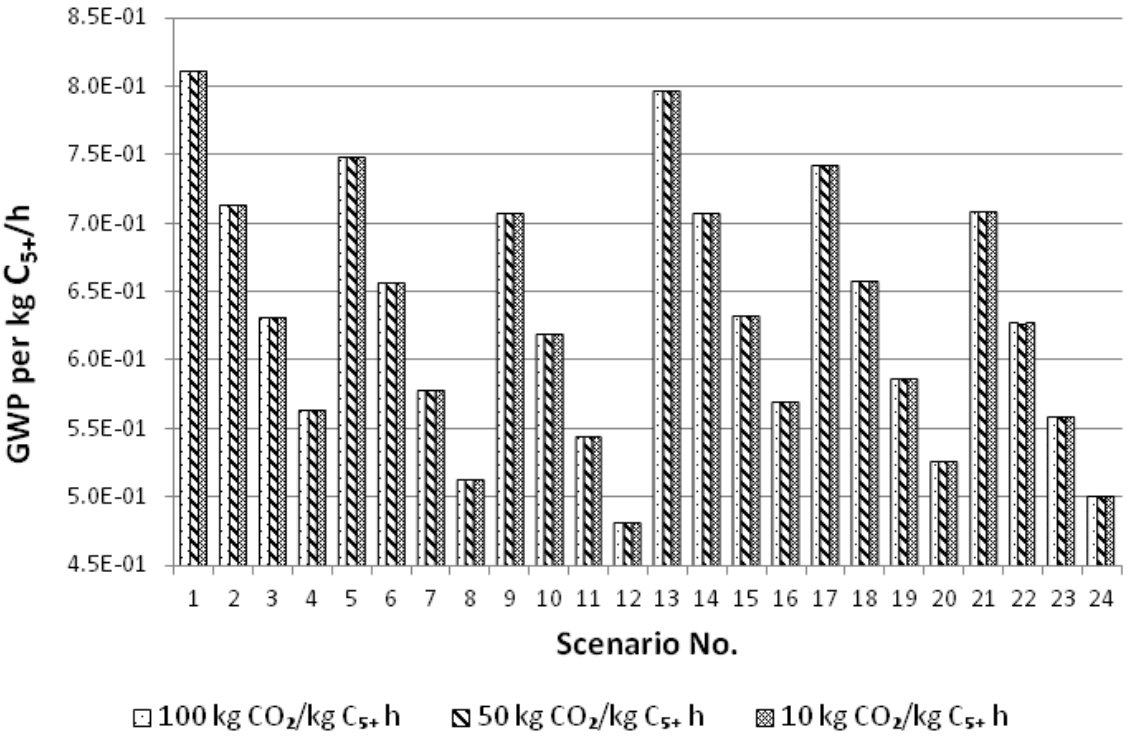

Figure 5.25: Normalized GWP results for three reactor sizes for scenario 1 - 24 of the GtL process.

5.5.4 Discussion

The GtL process uses a simplified FTS model as described in Section 5.4.2.1 as well as a simplified model for steam reforming neglecting the WGS reaction due to the Aspen®Plus constraints. Allowing WGS within steam reforming in Aspen®Plus would have result in a unacceptable H_2:CO ratio, whereas neglecting WGS yields a H_2:CO ratio of 3.7, which is close to the recommended ratio of 3:1 (*Baerns*, 2008). The process parameters are not validated with experimental results or with other GtL models since no GtL models in micro reaction engineering are available.

5.6 Offshore application of the GtL process

5.6.1 Introduction

As described in Section 2.3.6, one promising application field for 'micro' GtL plants are remote oil production and processing facilities offshore with the aim to replace gas flaring. Flared gas consists of so called green house gases, such as carbon dioxide and methane. Such replacement would result in avoiding green house gas emissions during exploration and facilitate the use of the associated gas. Thereby, a lowering effect to climate change can be expected. The avoidance of green house gas emissions, however, is only accurate under the premise that the green house gas emissions during entire life cycle of the microreactor are lower than the gas flaring volume, while excluding further use of the produced fuel. Since the boundary conditions vary considerably within the production and processing facilities in terms of usable gas volume per time, GtL plant scaling and design should be adapted. In Figure 5.26 the product system to be considered here is illustrated.

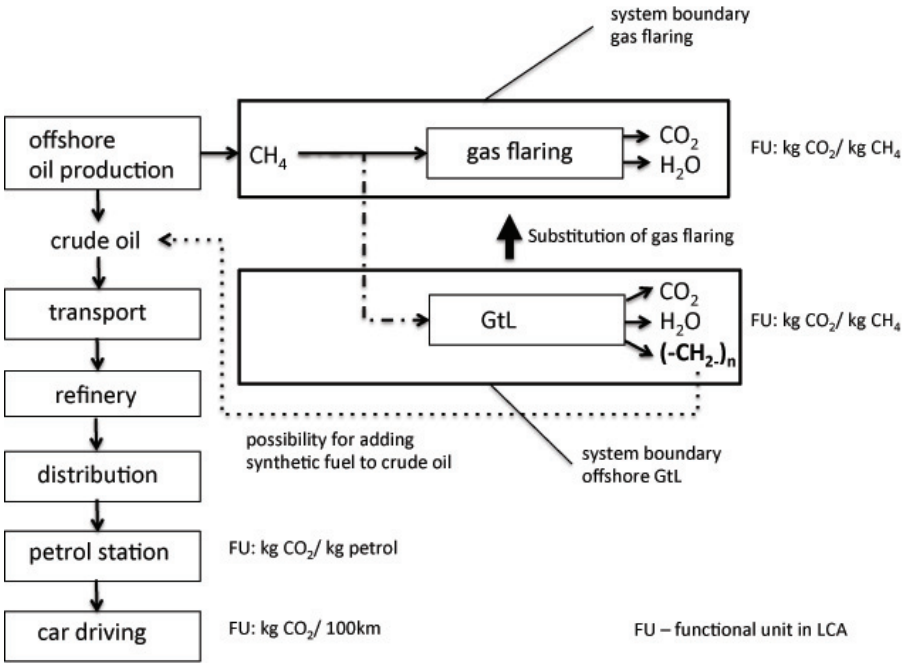

Figure 5.26: System boundaries for gas flaring and offshore GtL.

Despite GWP reduction, profit maximization plays an important role in oil industry and must be taken into account as well.

Within this section a optimization model is introduced for comparing different GtL plant designs in terms of GWP and costs optimization on the example of an offshore oil platform in the North Sea.

5.6.2 Methodology

5.6.2.1 R&D model

The GtL plant is described by the design dependent parameters production volume in kg h^{-1}, minimum capacity utilization rate μ_{min}, manufacturing costs $k_{Manufacturing}$(in Euro per GtL plant) and GWP impact in manufacturing $GWP_{Manufacturing}$ and use GWP_{Use} of the GtL plant. The optimization of one parameter is in general only possible by changing other parameters to the worse. For this reason, optimizing the whole GtL plant implies finding the best configuration of the considered parameters.

Since the overall model of the microreactor development follows the bottom-up approach, reactor respectively plant designs defined by specific parameters should be compared according to their appropriateness for the application field offshore. This type of mathematical problem can be solved by linear programming (*Grötschel*, 2004, *Matousek*, 2007, *Griva et al.*, 2009) for each parameter followed by weighting the results with respect to the desired goal.

Symbol	Parameter	Unit
a	product-reactant ratio	$kg_{product}\ kg^{-1}{}_{reactant}$
b_i	GtL plants in operation at time t_i	unit
b_{max}	maximum GtL plants in operation at time t_i	unit
$gas(t_i)$	associated gas volume at time t_i	kg
$k_{Manufacturing}$	manufacturing costs	Euro
m_i	converted associated gas at time t_i	kg
m^{max}	capacity GtL plant	kg
$p_{CrudeOil}$	crude oil price	Euro per kg (ρ = 0.85)
p_{CO_2}	CO_2 certificate price	Euro
$GWP_{Manufacturing}$	GWP in manufacturing process	kg CO_2-Eq.
GWP_{Use}	GWP in GtL process	kg CO_2-Eq.
$GWP_{Reduced}$	reduced GWP compared to gas flaring	kg CO_2-Eq.
μ_{min}	minimum capacity utilization rate	$kg\ kg^{-1}{}_{max}$

Table 5.9: Cost and GWP optimization parameters.

The cost and GWP optimization problems for offshore GtL plants are described in the following paragraphs. All used parameters for the optimization problems are summarized in Table 5.9.

Cost optimization

The offshore oil platform is defined by a specific associated gas volume at a discrete time-point t_i. Finding the optimized number of GtL plants b_i with specific design parameters, i.e. number and size of microreactors operating independently, for a specific gas volume gas at time-point gas_{t_i} with simultaneously maximizing the refund compared to gas flaring costs is the aim of this cost optimization. Assumptions for the optimization are listed below:

- GtL plants behave identical in terms of production volume, μ_{min}, $k_{Manufacturing}$, GWP_{Use}
- Defined number of operating GtL plants b_i at time-point t_i on the platform
- Operating and transportation cost of the GtL plants to the oil platform are neglected
- Refund is defined by the amount m_i of produced C_{5+} multiplied with the crude oil price $p_{CrudeOil}$ and the product/reactant ratio a
- Gas flaring costs are defined by the amount of flared gas at time t_i ($gas(t_i)$) multiplied with the CO_2 certificate price
- Manufacturing costs are defined by the maximum number of GtL plants in operation b_{max} b_i multiplied with the manufacturing cost of one plant $k_{Manufacturing}$

Equation 5.22 formalizes the linear-integer optimization problem of the cost optimization as described above.

$$\begin{aligned} max. \quad & \Delta G(m_1, m_2, ..., m_i, b_1, b_2, .., b_i) = \\ & (p_{CO_2} + p_{CrudeOil} \cdot a) \cdot \sum m_i - k_{Manufacturing} \cdot b_{max} \end{aligned} \tag{5.22}$$

GWP optimization

Overall goal of the GWP optimization is minimizing the GWP by using GtL plants. Therefore the aim is to find the most appropriate GtL plant design defined by specific parameters for a specific amount of associated gas. It is assumed that GWP during oil production is reduced by converting associated gas (methane) into liquid hydrocarbons $GWP_{Reduced}$ m_i within the defined system boundaries compared to gas flaring as illustrated in Figure 5.26. Beside the GWP reduction by converting associated gas, greenhouse gases

are produced during manufacturing $GWP_{Manufacturing}$ and use GWP_{Use} of the GtL plant. Equation 5.23 shows the objective function of the linear-integer optimization.

$$\begin{aligned} & max. \quad \Delta G(m_1, m_2, ..., m_i, b_1, b_2, .., b_i) \\ = & (GWP_{Reduced}) \cdot \sum m_i - GWP_{Use} \cdot \sum b_i - GWP_{Manufacturing} \cdot b_{max} \end{aligned} \tag{5.23}$$

Boundary conditions for Cost and GWP optimization

Cost and GWP optimization require the same four boundary conditions. The converted amount of gas m_i is equal or less than the associated gas volume gas(t_i) at time-point t_i as stated in the first boundary condition. In addition, gas m_i is equal or less than the capacities of the operating plants $b_i m^{max}$ (2nd boundary condition) and each operating plant should reach the minimum capacity as defined (3rd boundary condition). The fourth condition states that the amount of plants at the platform b_i is equal or less than the maximum amount of plants b_{max}.

$$\begin{aligned} s.t. \quad m_i &\leq gas(t_i) \\ m_i - b_i \cdot m^{max} &\leq 0 \\ -m_i + m^{max} \cdot \mu_{min} \cdot b_i &\leq 0 \\ b_i - b_{max} &\leq 0 \\ b_1, b_2, b_i, b_{max}, m_1, m_2, m_i &\geq 0 \\ b_1, b_2, b_i, b_{max} &\in \mathbb{N}_0 \end{aligned} \tag{5.24}$$

5.6.2.2 Parameter values

The GWP for manufacturing a whole GtL plant is assumed to be in the range of the GWP of one microreactor using chemical etching as structuring method (Design B, see Appendix B). This simplification was done for three reasons: 1) this work focuses mainly on the manufacturing of a FTS microreactor, 2) the considered GWP value determine the minimum since only one FTS reactor is considered in the GtL plant and 3) focus of this thesis is the proof of concept for the methodology and not the LCA results.

The GtL plant lifetime and thus the microreactor lifetime is assumed to be 30 years according to the average lifetime of an oil platform. Since the catalyst contribution to

the GWP is minor and recycling is proposed, only the initial catalyst loading is considered. Price for crude oil is set to 0.7311[1] Euro per kg and for CO_2 certificates to 0.0071[2] Euro per kg according to stock exchange values from December 2011. The product/reactant factor for the smallest plant scale is 0.69 and for the others 0.70, based on results of scenario 1 of the GtL process. Utilized values for GWP are listed in Table 5.10 and the GtL plant prices in Table 5.11. This GtL plant prices are based on the steel prices multiplied with a scaling factor for other required plant components.

Reactor scale	$GWP_{Manufacturing}$ in kg CO_2-Eq.	GWP_{GtL1} in kg CO_2-Eq.	GWP_{GtL2} in kg CO_2-Eq.
10 kg C_{5+} h^{-1}	26,420,562	74,663	66,489
50 kg C_{5+} h^{-1}	132,102,780	343,287	302811
100 kg C_{5+} h^{-1}	264,205,553	685,820	632829

Table 5.10: GWP values for manufacturing FTS microreactor Design A Table B.1 in Appendix B and GtL1/2 process for scenario 1/2 see Table D.1 in Appendix D.

Reactor scale	$k_{Manufacturing_{min}}$ in Euro	$k_{Manufacturing_{max}}$ in Euro
10 kg C_{5+} h^{-1}	10,000	100,000
50 kg C_{5+} h^{-1}	50,000	250,000
100 kg C_{5+} h^{-1}	100,000	350,000

Table 5.11: GtL plant prices for three plant scales.

Gas flaring volume estimations are based on results of the GGFR project (*The World Bank*, 2002). Annual gas flaring volumes per oil platform are calculated by summarizing gas flaring volumes of the European countries owning oil platforms in the North Sea and dividing the total volume by the number of oil platforms counted visually from Google Earth® pictures (*National Geophysical Data Center (NOAA)*, 2011) per year for the years 1994 - 2010. Based on the seventeen annual values, the arithmetic mean (15,000 t a^{-1}), a minimum (9,000 t a^{-1}) and a maximum value (24,000 t a^{-1}) per platform are defined and utilized as associated gas volumes (gas t_i). For all simulations a linear declining gas distribution with at least 30 discrete time steps is implemented starting with the maximum value.

[1] Nymex Crude Future, http://www.bloomberg.com/energy/

[2] European Energy Exchange, http://www.eex.com/de/Marktdaten/Handelsdaten/Emissionsrechte/European

5.6.2.3 LCA Model

The LCA results for manufacturing (GWP per microreactor) from Section 5.3 and the use of the microreactor (GWP in kg CO_2-Eq. $kg^{-1}_{C_{5+}}$) from Section 5.5 are used as input values $GWP_{Manufacturing}$ and GWP_{Use} for the optimization and therefore no additional LCA model is necessary. This input value is also called interface parameter according to the MSCS methodology.

5.6.2.4 Interfaces

$GWP_{Manufacturing}$ is an interface parameter between the modules *Manufacturing* and *Offshore application* and GWP_{Use} between the modules *GtL* and *Offshore application*.

5.6.2.5 Software

Matlab®R2011b is used for the implementation of the linear-integer optimization algorithm, and Gurobi® 5.0.0 as linear programming solver. The program code can be found in Appendix E.

5.6.3 Results

Having the same amount of associated gas available, the GtL plant with 50 and 100 kg C_{5+} h^{-1} receive with standard crude oil price in average 2 % higher refund during their lifetime compared to the 10 kg C_{5+} h^{-1} as shown in Figure 5.27 and 5.28.

Varying the crude oil price by -/+ 50% decrease or increase refund by 50 %, whereas varying the CO_2 certificate price in the same way only yields $\pm$0.7 % refund. Reason for this is the price difference between crude oil and CO_2 certificates (factor 100).

When increasing the GtL plant price for all plant scales according to Table 5.11, the refund is reduced by 11 % for 10, 13 % for 50 and 3% for 100 kg C_{5+} h^{-1} plants.

The highest CO_2 savings are achieved for the 100 kg C_{5+} h^{-1} GtL plant with the operating conditions of GtL1, whereas with GtL2 the GtL plant with 50 kg C_{5+} h^{-1} shows the best performance in terms of CO_2 savings as seen in Figure 5.30. The smallest GtL plant performed worst for all conditions.

In summary, the 50 kg C_{5+} h^{-1} GtL plant with GtL2 operating conditions shows the best performance for the incurred associated gas volume for refund maximization and GWP minimization.

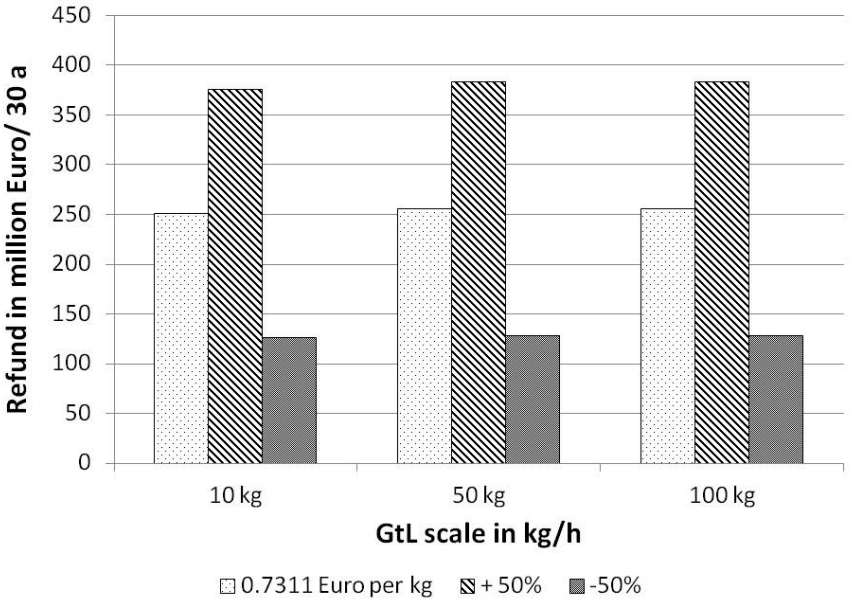

Figure 5.27: Influence of the oil-price variation.

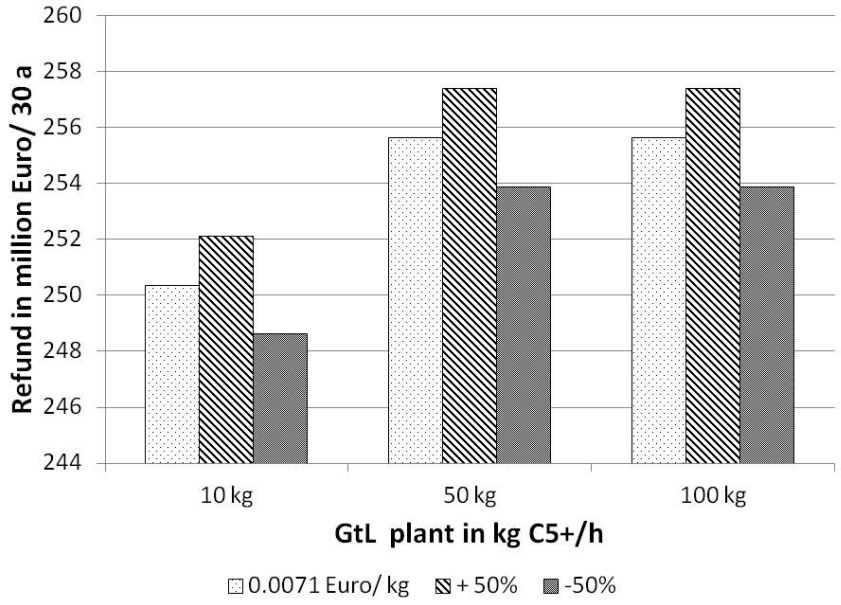

Figure 5.28: Variation in CO_2 certificate price.

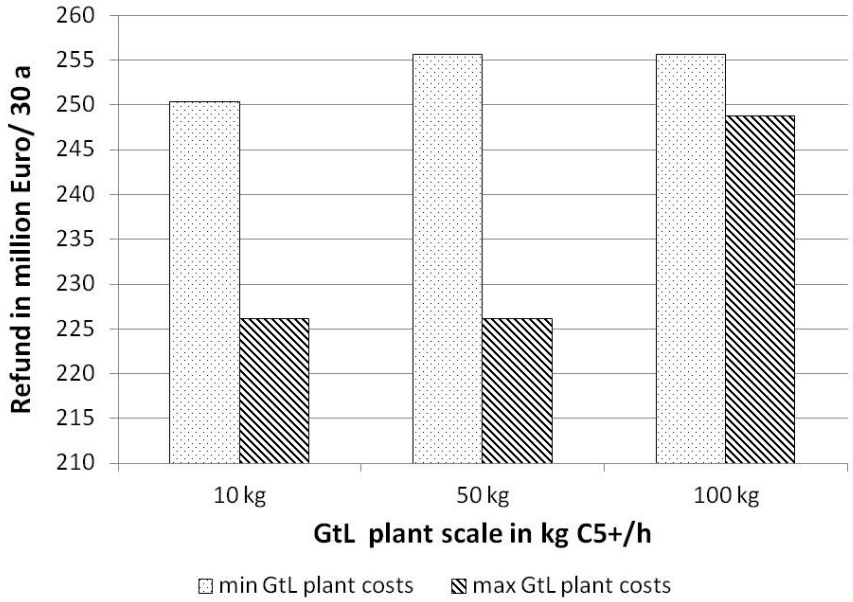

Figure 5.29: GtL plant cost variation.

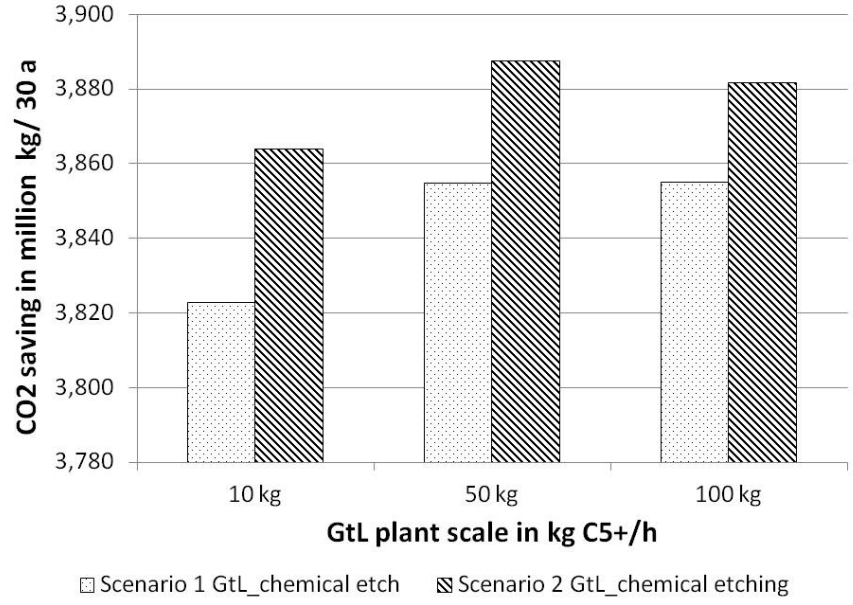

Figure 5.30: GtL plant scaling for two GtL process configurations (scenario 1 and scenario 2).

5.6.4 Discussion

The optimization module compares different GtL plants in terms of GWP savings and refund maximization under given boundary conditions such as associated gas volume, plant size, and CO_2 certificate price. Since the performance strongly depends on the plant design, which determines the price and the GWP in manufacturing and use (see results in Section 5.3.3 and 5.5.3), uncertainties of this values strongly effect the results of the optimization.

5.7 Systems analysis

In Section 5.3 to 5.6 single horizontally modules within the whole development process of a microreactor for a specific application field were examined according to the proposed interests. The modules belong to different system hierarchies and represent different steps in time within the microreactor development process. In this section the modules are analyzed according to three different point of views of a systems analysis: their interdependencies in vertical system hierarchy representing time dependent system complexity (Section 5.7.1), the impact of the manufacturing and use phase of a microreactor from a life cycle perspective (Section 5.7.2), and finally their relevance to the GWP impact in the offshore application field to replace gas flaring (Section 5.7.3).

5.7.1 Microreactor development

The vertical system hierarchy interdependencies, representing the microreactor development, are analyzed following the bottom-up approach, which means from low level to higher level system hierarchy. For each level relevant parameters for the system performance are determined. Moreover, parameters of interfaces to other system hierarchies are determined. This allows back tracing of system performance results to lower level hierarchy parameters as explained in Section 4.3.1.

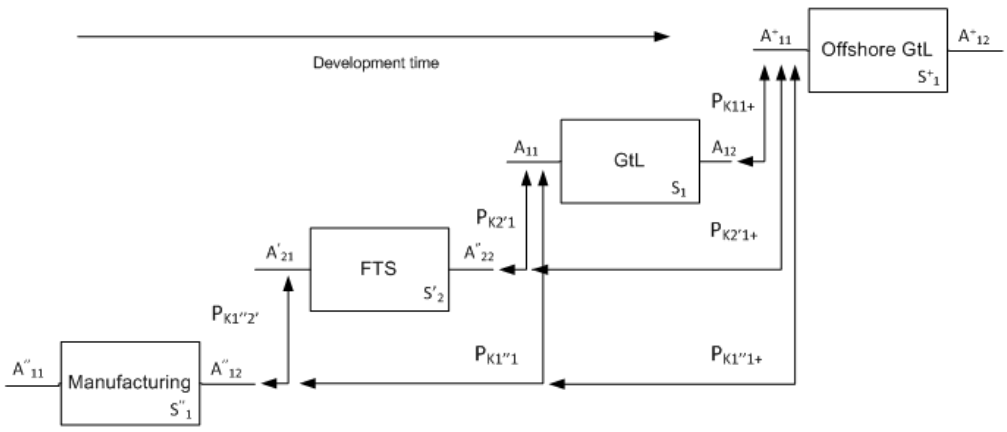

Figure 5.31: Relations between hierarchical systems of the microreactor development. Symbols: system (*S*), in- and output attribute (*A*), relation between hierarchical systems (*P*).

In Figure 5.31 the relations between hierarchical systems of the microreactor development are illustrated with respect to the formal systems definitions in Appendix A and the modularization of the microreactor development in Section 5.2. Interdependencies between

the subsystems of the module *Manufacturing* are already analyzed in Section 5.3.2. Thus, the module *Manufacturing* is determined as the lowest system hierarchy for the vertical systems hierarchy analysis.

The module *Manufacturing* is defined by parameters[3] of the microreactor design (e.g. catalyst void fraction) and the design dependent operation conditions for the FTS synthesis (selectivity, productivity, temperature, pressure). Until now, experimental results allowing to validate the design dependent FTS performance are lacking. For this reason, the scenario analyses in the module *FTS* were performed with FTS parameter variations for one microreactor design, see Section 5.4. So far, results from Section 5.3 and Section 5.4 indicate that the design parameter *catalyst void fraction* mainly determines the GWP in *Manufacturing* and the parameter *selectivity* determines the GWP performance in *FTS*. Hence, the actual correlation between these two parameters should be validated experimentally.

The module *FTS* is defined by the FTS operation parameters (selectivity and chain growth probability, productivity, temperature, pressure, conversion rate), which are interface parameters to the module *GtL*. Due to a lack of experimental results towards a correlation between these FTS parameters, scenario analysis was performed to obtain information about the impact of three relevant *GtL* parameters (chain growth probability, conversion rate and pressure) on the GWP, see Chapter 5.5. Results indicate that the GWP performance strongly relates to the chain growth probability and therefore to the selectivity. Increasing the chain growth probability from 0.84 to 0.86 (CR=0.7, 20 bar) yields the same reduced GWP as increasing the conversion rate from 70 to 90 % (α = 0.84, 20 bar).

In offshore application, relevant parameters are GtL-plant cost and the GWP impact in manufacturing and use. Since GtL-plant costs depend on the costs for the raw material of the microreactors, while FTS catalyst costs can be regarded as insignificant, the costs ultimately depend on the microreactor design. In other words, high catalyst void fraction in microreactor design requires less raw material and therefore results in lower GtL-plant costs. Similarly; high catalyst void fraction results in lower GWP in *Manufacturing*, see Section 5.3.3.4. The GWP impact in the use phase mainly depends on the parameter selectivity (or chain growth probability), which is determined by the operation conditions in the microreactor, which in turn is determined by the microreactor design.

Based on these observations, the design phase of the microreactor development must be regarded as the most important phase for the later performance of the microreactor both, as a single component or as part of other technical systems, as assumed in Chapter 1.

[3]Parameters are called attributes in the formal systems definition

5.7.2 Manufacturing and use phase of a microreactor

The environmental impact of the microreactors life cycle phases is used to gain information about the relevance of each life cycle phase. Based on these results the relevance of the products life time can be estimated, which have to be taken into account in the microreactor design, i.e. catalyst lifetime, catalyst filling methods.

Focus is laid on the life cycle phases *Manufacturing* and *Use*, because a high impact of the manufacturing phase is to be expected for new technologies (*Kunnari*, 2009). However, recycling can be ignored here as established recycling methods exist for both main raw materials, steel and catalyst.

Design parameters for the manufacturing phase are taken from design B (see Appendix B Table B.1) for both structuring processes, and for the GtL process as use phase from scenario 1 (see Appendix D Table D.1) to represent first experimental results from a FTS microreactor (*Myrstad et al.*, 2009). The annual associated gas volume is set to 9 million kg a^{-1} (minimum value for oil platforms in the North Sea, see Section 5.6.2.2). Because the focus is put on the FTS microreactor development and the lack of additional data, the manufacturing phase is represented by the manufacturing of one microreactor, neglecting other components and reactors of the GtL system.

Figure 5.32 and 5.33 show the GWP impact in kg CO_2- Eq. per kg product (C_{5+}) for different reactor life times for three reactor sizes (10, 50, 100 kg h^{-1}) and two structuring methods in manufacturing. Reducing the reactor lifetime down to one year increases the GWP impact per kg product significantly for chemical etching as structuring method compared to a moderate increase for micromechanical machining. The decline of the GWP with increasing reactor lifetime suggests that the GWP impact during manufacturing is much higher than during the use of the microreactor, e.g. using the GtL plant for 1 day produces less emissions than the manufacturing of the plant. Since chemical etching in microreactor manufacturing has a higher impact than micromechanical machining (see Section 5.3.3.1), this conclusion also explains the moderate GWP reduction of micromechanical machining compared to chemical etching with increased lifetime.

Further results show that the 10 kg reactor size has a higher impact than the 50 and 100 kg reactor size for the considered gas volume when using micromechanical machining as structuring method in manufacturing. This can be explained by the higher number of reactors (GtL plants) for the conversion of the considered gas volume, and therefore the higher impact in manufacturing, e.g. one 50 kg h^{-1} reactor has a lower GWP in manufacturing than five 10 kg h^{-1}. Effects of chemical etching are less prominent but similar to those of micromechanical machining; increasing the reactor size, reduces the GWP per kg product

for all considered reactor life times.

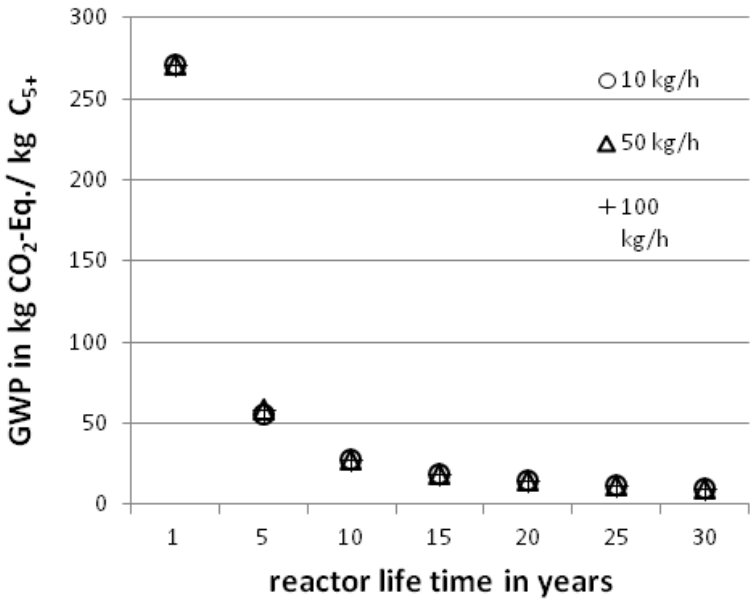

Figure 5.32: GWP change with reactor life time (chemical etching as structuring process).

In conclusion, the life cycle phase *Manufacturing* manifests a significantly stronger impact on GWP than the use phase of a microreactor for a reactor life time less than 5 years. Nevertheless, both life cycle phases do impact the ultimate GWP. Thus, the extension of the microreactor and GtL-plant life time represent an important design criteria for an environmentally optimized design. Based on current development phase of the FTS microreactors and available knowledge, the catalyst lifetime and catalyst integration methods may be a critical factor for the lifetime extension.

5.7.3 Application of microreactors to replace gas flaring

In this final section I will address the original question of this study to identify the optimal microreactor design and GtL process for the substitution of gas flaring in oil production under an ecological perspective. In combination with associated gas as feedstock, microreactors show high potential for decentralized fuel production. In order to replace gas flaring, decentralized fuel production would reduce greenhouse gas emissions during exploration and facilitate the use of associated gas. Thereby, a lowering effect to climate change can be expected. The reduction of greenhouse gas emissions, however, is only accurate under the premise that greenhouse gas emissions during the entire life cycle of the microreactor are lower than gas flaring emissions, while excluding further use of the pro-

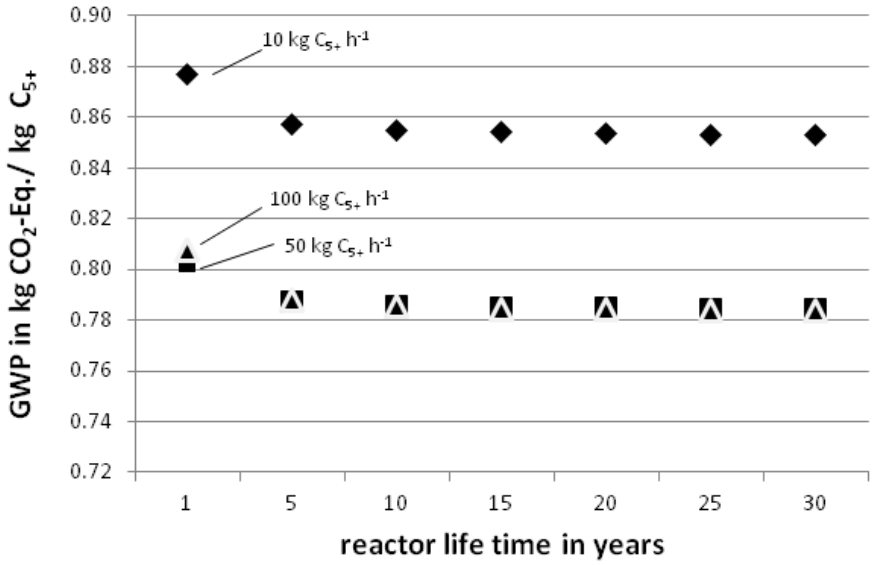

Figure 5.33: GWP change with reactor life time (milling as structuring process).

duced 'by-product' fuel. Therefore, the GWP emissions are compared for gas flaring and GtL processes using a specific associated gas volume. In addition, gas venting is included representing the worst case scenario for using associated gas in oil production. In Figure 5.34 three manufacturing processes for three microreactor sizes are compared with gas flaring and venting for the minimum (9 million kg a^{-1}) and maximum value (24 million kg a^{-1}) of gas per oil platform in the North Sea, see Section 5.6.2.2.

For the sake of simplification, it is assumed that the total gas volume consists of methane while other emissions related to gas flaring are neglected, i.e. technology for conversion of methane into synthesis gas. The CO_2-Eq. for methane is 21 (*United Nations*, 1995), which is equivalent to the value used in GWP100 (*Guinee*, 2001a,b). Equation 5.25 shows the chemical reaction for the full oxidation of methane (gas flaring). Furthermore, the use phase of the microreactor can be neglected, because of the low GWP impact (see Section 5.7.2). Thus, the GtL process is defined by the FTS microreactor design and manufacturing.

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O \tag{5.25}$$

Since CO_2 emissions in manufacturing only consider one FTS microreactor rather than an entire GtL plant, the premise that greenhouse gas emissions during the life cycle of a microreactor are lower than gas flaring is only accurate, if the lifetime of the GtL-plant is extended to significantly more than one year and mechanical micromaching is used in the

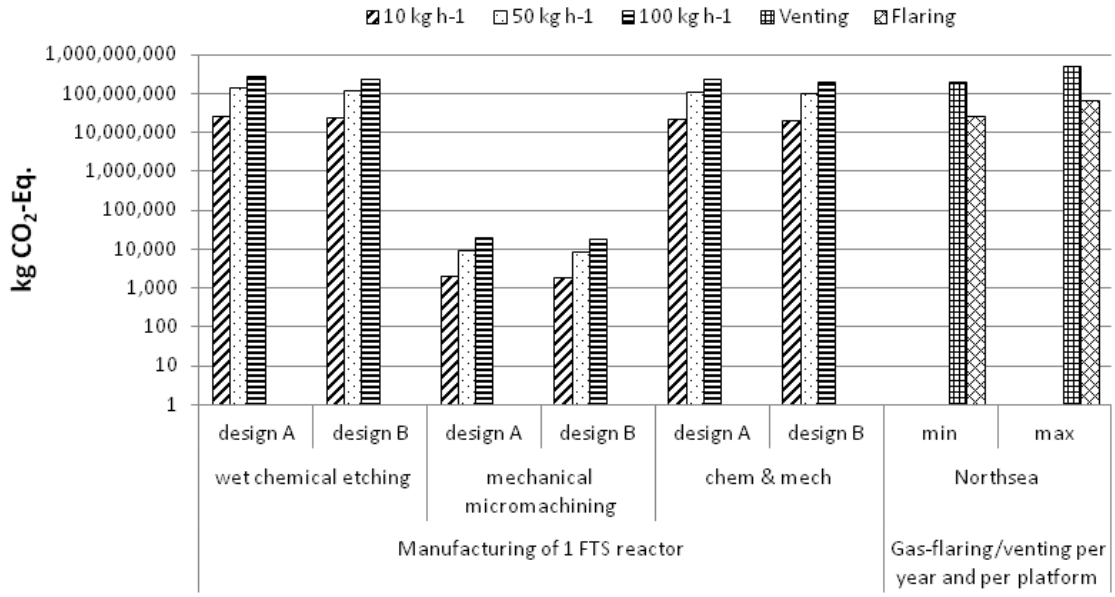

Figure 5.34: GWP for three GtL process scales in offshore application converting 9 million kg methane per year compared to gas flaring and venting. These GtL processes are described by the designs and the manufacturing technique of the considered microreactor.

manufacturing process.

In conclusion, the microreactor with design B (Co catalyst, 1.9 g g^{-1} h^{-1} productivity, 0.5 catalyst void fraction, see also Table B.1), fabricated by micromechanical machining and a reactor scale of 10 kg C_{5+} h^{-1} is the most appropriate for the application field from an environmental point of view. This design shows the least environmental impact (CO_2 emissions) for the considered boundary conditions in the application field compared to the other design possibilities.

5.7.4 Discussion

In Section 5.3 to 5.6 single modules of the microreactor development were analyzed with respect to the interest of the involved stakeholders and decision makers using LCA analysis with the new MSCS-methodology. The system analysis undertaken in Section 5.7 is based on the analysis results of the single modules of the microreactor development. Thus, the reliability of these results depends on the reliability of the results of the single modules, see Section 5.3.4, 5.4.4, 5.5.4, and 5.6.4.

CHAPTER 6

Discussion

The aim of this thesis was to develop an appropriate methodology for the analysis of interdependent device and process development from a life cycle perspective by applying state of the art concepts for interdependent systems modeling to LCA. Focus was put on the early research phase in technology development where the degree of freedom and therefore the impact of decision-making is higher than in later stages.

The Modular Server-Client-Server (MSCS) methodology, developed in this thesis, was validated on the example of Gas to Liquid (GtL) fuel conversion via Fischer-Tropsch synthesis (FTS) in microreactors, and first results are published in (*Zschieschang et al.*, 2012a,b).

My research offers the following significant contributions:

1. Scientific approach for modeling rebound effects and dynamic approaches in LCA
2. LCA analysis in early R&D of technologies, characterized by time dependent technology complexity in the development phases
3. GtL process in micro reaction engineering from a life cycle perspective

Scientific approach for modeling rebound effects and dynamic approaches in LCA

'The concept of rebound effects has been treated by many authors, but its applicability in LCA is still an open question, because of the complexity, uncertainty and costs involved.'(*Zamagni*, 2006, p.7). By using the general systems theory and the MSCS methodology for modeling in LCA, rebound effects and complexity are now illustrateable in a consistent scientific framework.

Since *'dynamic approaches are still pioneering, and several efforts are still necessary both at methodological and practical level'*(*Zamagni*, 2006, p.8) and *' the available software tools do not reflect advances in modeling, because they are based on static relations'*(*Zamagni*, 2006, p.8), the MSCS methodology with the general systems theory offers a practicable basis for dynamic approaches on a theoretical and practical level. By extending the definitions in the general systems theory via associating systems in different system hierarchies, the time dimension between independent systems (changing technology complexity in development phases) can be also described by the application of the methodology.

LCA analysis in early R&D of technologies

The aim for LCA in technology development is to receive information on how to design an environmentally friendly technology. Since process specific information is limited in early development phases of technologies, some authors suggest the use of simplified LCA (*Rydh*, 2005, *Hübschmann et al.*, 2009) and screening approaches (*Gasafi*, 2006). Compared to these approaches that use available data in technology development for a simplified LCI (system boundaries), LCIA (impact assessment methods) and screening of crucial parameters (pareto analysis), the MSCS methodology uses available scientific models (i.e. machine tool model) for foreground processes in technology development and translates them into LCA models (black box models) on the basis of the general systems theory. Consequently, the MSCS-methodology allows the design and life cycle phase specific environmental impact calculation with a high level of detail for the considered technology under development. Because of the combined LCA and R&D model, LCA results can be directly transfered back to the R&D process, which is not feasible with the other approaches.

Since no parameterized databases for machine tools and appropriate LCA software for the use of these data are available, the MSCS methodology is highly time and resource intensive compared to simplified and screening approaches. Nevertheless, research towards parameterized databases (*Overcash*, 2012, *Kellens et al.*, 2012) and appropriate software tools (*Hedemann*, 1998-2012) is under development and may be available soon.

GtL process in micro reaction engineering from a life cycle perspective

The MSCS methodology is a guideline on how to use LCA in technology development. Applying this new methodology requires various tools, i.e. software tools, R&D models

for foreground processes, parameterized data bases (machine tool models), which are not yet available or under development. Microreactor development for the application in a GtL system offshore is highly complex and requires various R&D models for the fabrication processes, software tools (i.e. Matlab®, AspenPlus®) and mathematical algorithms for the interdependencies between design criteria and operation conditions of a microreactor in a GtL plant. In addition, no GtL plant with microreactors exists. Because of this boundary conditions numerous simplifications have been applied in the validation of the MSCS methodology and thus, results require cautious interpretation. Obtained results indicate a strong environmental impact within the manufacturing of the technologies compared to the use phase. Because the GtL plant is designed as self-sustaining system recycling side products for heat generation of the endothermic reforming process, these results are realistic. Due to the combination of high level of detail modeling with modeling of different system hierarchies, single design parameter influencing the FTS, GtL-process and GtL offshore system performance could be identified in advance.

CHAPTER 7

Outlook & Conclusion

7.1 Outlook

The work of the thesis is seen as a first step towards a scientific LCA framework allowing the use of LCA analysis in different application fields. Besides further research in this field, the combination of LCA methodology developments in different application fields may be advantageous, i.e. combining machine tool modeling in life cycle engineering with parameterized database development in the LCA community as introduced in this thesis. For such combination two main objectives should be in focus:

Development of parameterized databases

The development of a new type of databases is ongoing in the LCA community, but still in its infancy. The term 'parameterized database' is not yet consistently defined. Using the MSCS methodology and the general systems theory as basic scientific approach, parameterized databases may be linked to machine tool modeling in manufacturing. This would allow a side specific, high level of detail LCA for manufacturing processes of the new technology which is advantageous in technology development, especially for new technologies expected to have a high environmental impact in this life cycle phase.

Development of flexible LCA software

Successful application of the MSCS methodology requires the availability of appropriate software tools. Further improvements are thus necessary, especially in the field of software interfaces allowing dynamic modeling by using state of the art software tools in technology development (Computer Aided Design (CAD), Computational Fluid Dynamics (CFD),

Matlab®, Aspen®Plus, Ansys®) with LCA software within one run time system. Moreover, LCA software should be designed allowing the use of parameterized databases and information flows.

7.2 Conclusion

The aim of this thesis was to analyze how to apply LCA for technology development. As a case study for interdependent device and process development, I chose the development of a microreactor for the FTS in a GtL process in the discipline of micro process engineering. Based on the results of the literature research towards state of the art LCA methodology in general and technology development in particular, as well as basic characteristics of the chosen case study in micro reaction engineering, requirements for the use of LCA in technology development were identified and formulated.

The focus of the theoretical work was put on the modeling of the microreactor development process and how to deal with different models in LCA and microreactor development. Based on the general systems theory, a model of micro-reactor development was created. Subsequently, a method for the application of LCA on the model of micro-reactor development was developed. This novel method is called Modular Server-Client-Cerver methodology. Using this novel methodology, four phases of the microreactor development were analyzed by LCA; design and manufacturing, use of a microreactor for the FTS, application of the microreactor in the GtL process and application of the GtL process for the substitution of gas flaring in offshore oil production. This method helped to chose, from an environmental point of view, the most suitable microreactor design for a GtL process replacing gas flaring in offshore oil production.

In conclusion, the developed MSCS-methodology allows early evaluation of different microreactor designs within different processes for specific application fields from a life cycle perspective. The herein established MSCS methodology enables a more sustainable product and process development by allowing to reduce the development time as well as costs and required resources. Furthermore, the methodology utilizes common research software-tools and therefore supports an easy integration in current device and process design in order to facilitate its acceptance and application in industry.

APPENDICES

APPENDIX A

Formal systems definition

This appendix presents a summarized mathematical system definition subdivided into fourteen single definitions (*Ropohl*, 1979). In (*Ropohl*, 2009) those definitions are developed further up to twenty-five. In this thesis the fourteen definitions from (*Ropohl*, 1979) are used, because they adequately describe the considered system and they did not change in content compared to (*Ropohl*, 2009).

$$S = (\alpha, \varphi, \sigma, \pi) \tag{A.1}$$

$$\text{with } \alpha = \{A_i\};\ \varphi = \{F_j\};\ \sigma = \left\{S_k^{'}\right\};\ \pi = \{P_m\}$$

$$\text{and i} \in \text{I; j}\in \text{J; k}\in \text{K; m}\in \text{M (I, J, K, M} \sim \mathbb{N})$$

If α is a set of attributes of A_i, φ a set of functions F_j, σ a set of subsystems $S_k^{'}$ and π a set of relations P_m than is the quadruple of the sets a system.

$$A_i = \{a_{in}\} \neq \emptyset \tag{A.2}$$

$$\text{with i}\in\text{I and n}\in\text{N (I,N} \sim \mathbb{N})$$

The attribute A_i is a not empty set of characteristics a_{in}.

$$\alpha_x \cap \alpha_y = \emptyset; \alpha_y \cap \alpha_z = \emptyset; \alpha_z \cap \alpha_x = \emptyset; \tag{A.3}$$
$$\alpha_x \cup \alpha_y \cup \alpha_z = \alpha;$$
$$\alpha_g \notin S$$

$$F_j \subset X_{i \in I} A_i \tag{A.4}$$

The function F_j is a proper subset of the cartesian product between the attributes A_i.

$$F_{xj} \subset XA_{xi}; F_{yj} \subset XA_{yi}; F_{zj} \subset XA_{zi}; \tag{A.5}$$

The function between the inputs A_{xi} is called input function F_{xj}, between the outputs A_{yi} output function F_{yj}, and between the status A_{zi} status function F_{zj}.

$$F_{Tj} : A_{zi} x A_{xi} \rightarrow A_{zi}; F_{Rj} : A_{zi} x A_{xi} \rightarrow A_{yi}; F_{Mj} : A_{zi} x A_{zi} \rightarrow A_{yi} \tag{A.6}$$

The function between inputs and status is called transfer function F_{Tj}, between inputs and outputs result function F_{Rj} and between status and outputs marker function F_{Mj}.

$$S_k^{'} = (\alpha_k^{'}, \varphi_k^{'}, \sigma_k^{'}, \pi_k^{'}) \tag{A.7}$$

The subsystem $S^{'}$ is a system in accordance with the definition A.

$$P_m \subset X_{k \in K} A_{ki}^{'} \tag{A.8}$$

with one i per k

If $A_{ki}^{'} \in \alpha_k^{'}$ is a attribute of the subsystem $S_k^{'}$, than the real subset of the Cartesian product between k attributes of k subsystems is called a k-diget relation P_m. The set π of the relation is called structure of the system S.

$$P_{K12} \subset A'_{1y} x A'_{2x} \tag{A.9}$$

$$\text{with } a'_{2xn} = a'_{1yn} \text{ for all n}$$

If A'_{1y} is the output of subsystem S'_1 and A'_{2x} the input of subsystem S'_2 than the double-digit identity relation between A'_{1y} and A'_{2x} is called interconnection P_{K12}.

$$\gamma = \beta / \sigma \tag{A.10}$$

β is a not empty set, containing the set $\sigma \subseteq \beta$ of the subsystems as subnets. Therefore, the difference set between set β and set σ of the subsystem surrounding area is called γ.

$$S^+ = (\alpha^+, \varphi^+, \sigma^+, \pi^+) \tag{A.11}$$

$$\text{with S} \in \sigma^+$$

If σ^+ is a system set containing the system $S \in \sigma^+$, than the quadruple containing σ^+ is called supersystem S^+ to system S according to A

$$H = (..., S'', S', S, S^+, S^{++}, ...) \tag{A.12}$$

The sequence of system S, subsystems S', S'', etc. and supersystems S^+, S^{++}, etc. is called system hierarchy.

$$S_b, S_{b-1}, S_{b+1} \tag{A.13}$$

The ordinal number of a system is called system hierarchy with b$\in \mathbb{N}$ of system S, b-1 for subsystem S' and b+1 for supersystem S^+.

$$S = S_{real} \Leftrightarrow (\alpha \neq \phi) \wedge (\sigma \neg \phi) \wedge (\gamma \neg \phi) \tag{A.14}$$

The system S is called real system if the set of attributes α, subsystems σ and surrounding area γ are not empty sets.

APPENDIX B

Manufacturing

This appendix includes modeling parameters, data flow diagram for the Manufacturing model, used result processes of Ecoinvent v2.2. and Umberto® scripts.

1 Design and manufacturing parameter

1.1 Ecoinvent data

Result processes from Ecoinvent v2.2 (*Weidema and Hischier*, 2010):

- aluminium oxide, at plant [RER]
- cobalt, at plant [GLO]
- electricity, production mix DE [DE]
- iron(III) chloride, 40% in H_20, at plant [CH]
- light fuel oil, at regional storage [RER]
- nickel, 99.5%, at plant [GLO]
- nitric acid, 50% in H_20, at plant [RER]
- steel, electric, chromium steel 18/8, at plant [RER]
- tap water, at user [CH]
- water, deionized, at plant [CH]

1.2 Design/Scenario parameters

Design Scenario	Catalyst	Temperature T in °C	Productivity in g g^{-1} h^{-1}	Pressure p in bar	Catalyst void fraction	Bulk density in kg m^{-3}	Loss cut	Welding loss factor	Abrasion factor
A	Co/Ni	225	1.7	20	0.5	1000	0.2	0.2	2
B	Co	225	1.9	20	0.5	1000	0.2	0.2	2
C	Co/Ni	225	1.7	20	0.4	1000	0.2	0.2	2
D	Co/Ni	225	1.7	20	0.6	1000	0.2	0.2	2
E	Co	225	1.7	20	0.5	1000	0.2	0.2	2
F	Co	225	2.1	20	0.5	1000	0.2	0.2	2
G	Co	225	1.9	20	0.4	1000	0.2	0.2	2
H	Co	225	1.9	20	0.6	1000	0.2	0.2	2
I	Co	225	1.9	20	0.5	1100	0.2	0.2	2
J	Co	225	1.9	20	0.5	900	0.2	0.2	2
K	Co	225	1.9	20	0.5	1000	0.3	0.2	2
L	Co	225	1.9	20	0.5	1000	0.1	0.2	2
M	Co	225	1.9	20	0.5	1000	0.2	0.3	2
N	Co	225	1.9	20	0.5	1000	0.2	0.1	2
O	Co	225	1.9	20	0.5	1000	0.2	1	2
P	Co	225	1.9	20	0.5	1000	0.2	0.2	10
Q	Co	225	1.9	20	0.5	1000	0.2	0.2	20

Table B**.1:** Microreactor design parameters for the manufacturing processes.

2 Data flow analysis technology development

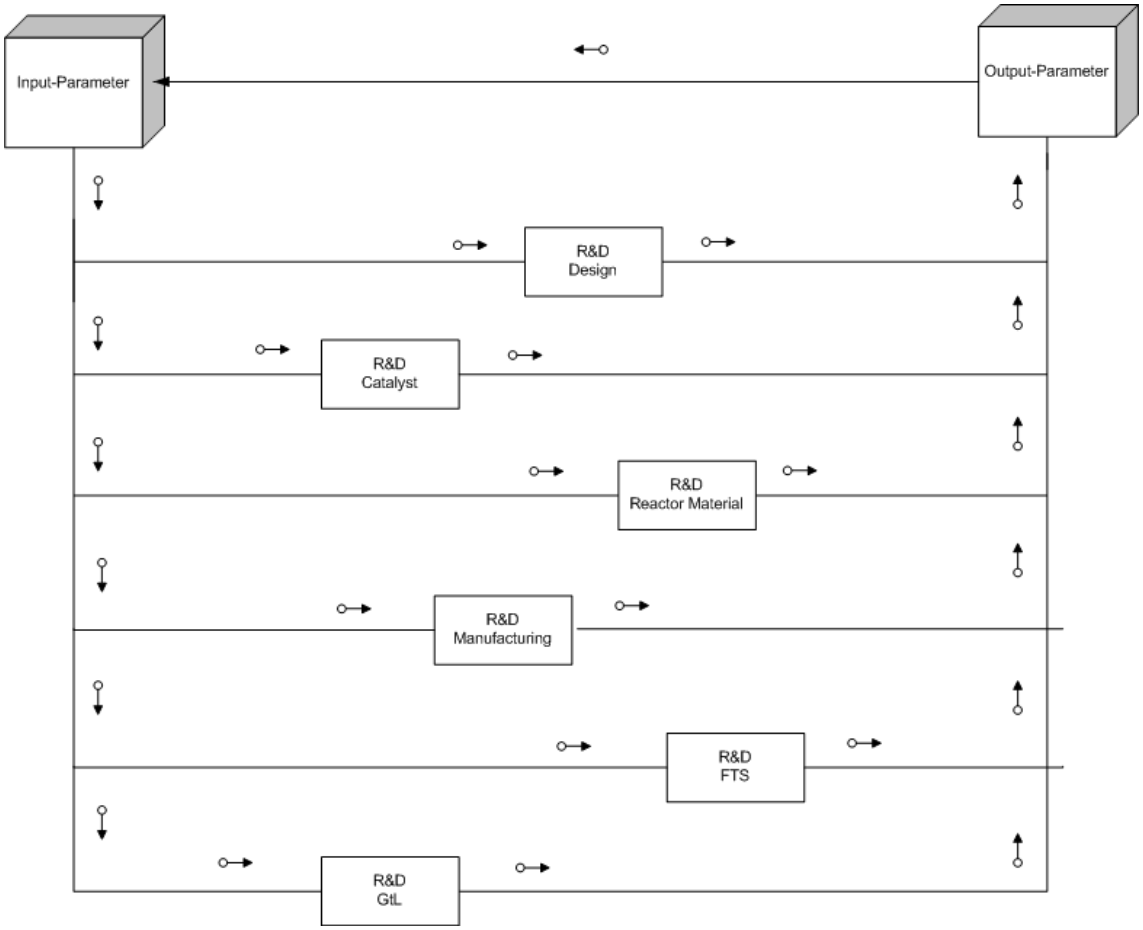

Figure B.**1:** Data flows between R&D modules in microreactor development.

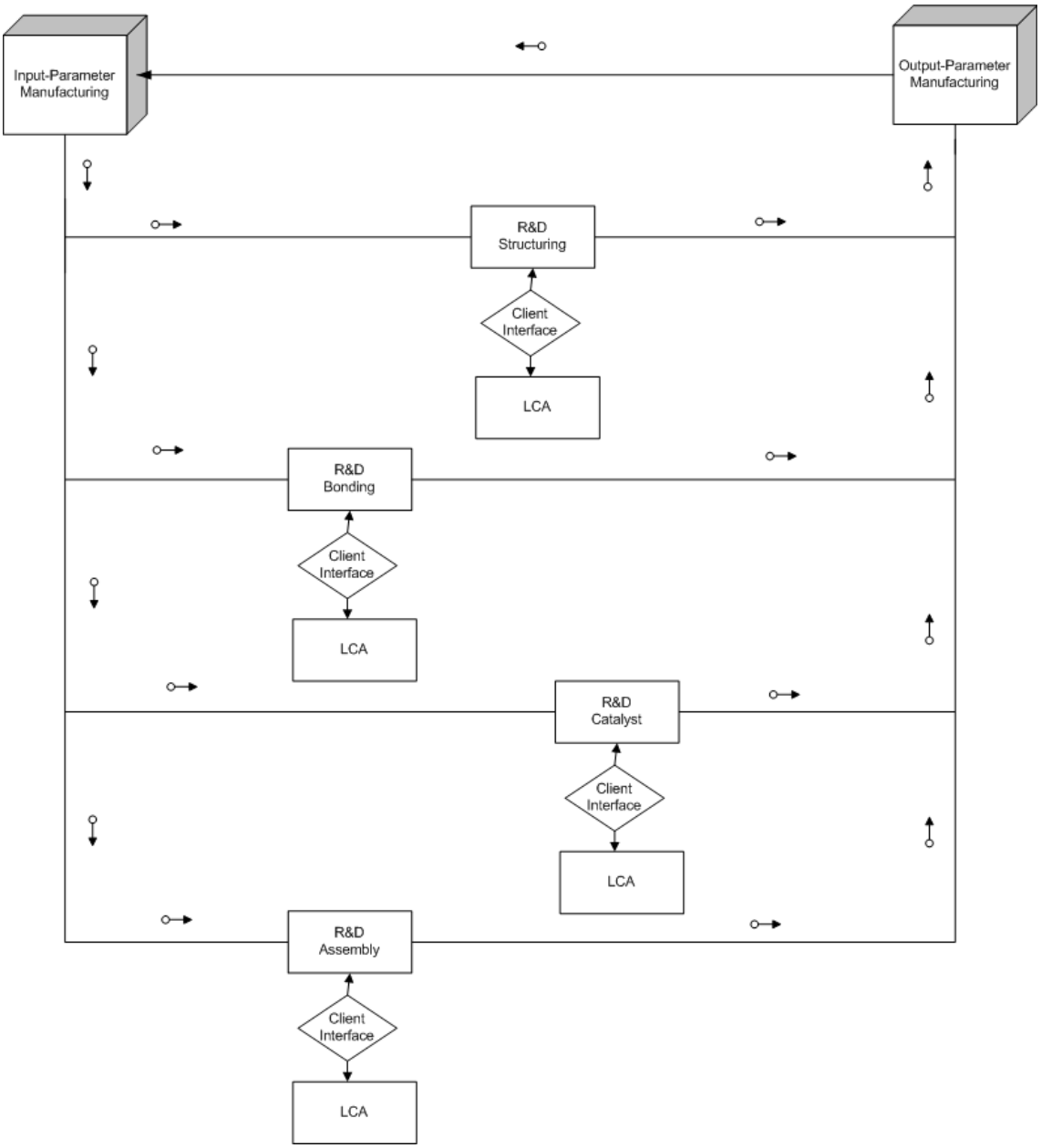

Figure B.**2:** Data flows within manufacturing module.

3 LCA modeling in Umberto®

3.1 Umberto network

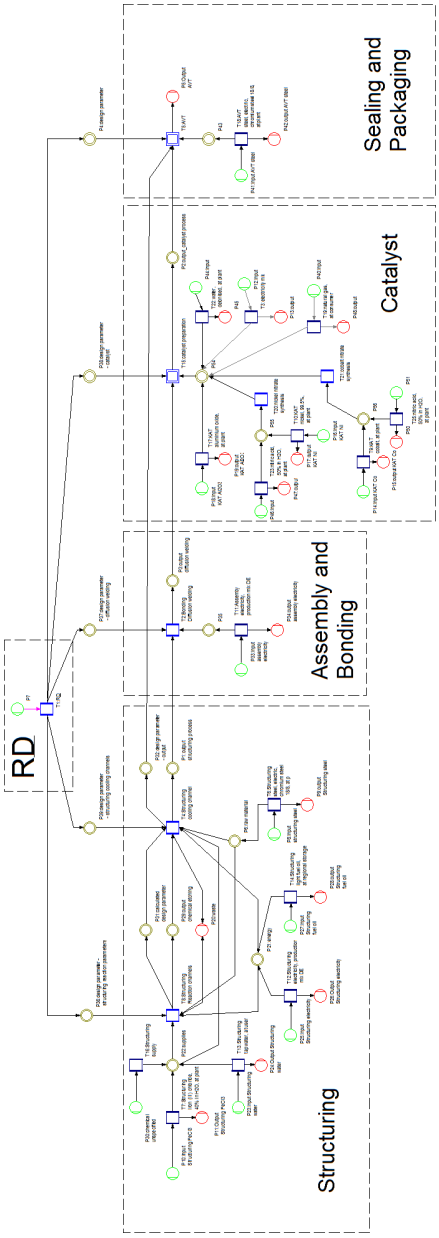

Figure B.**3:** Network of all manufacturing processes using chemical etching for reaction and cooling channel structuring in Umberto®.

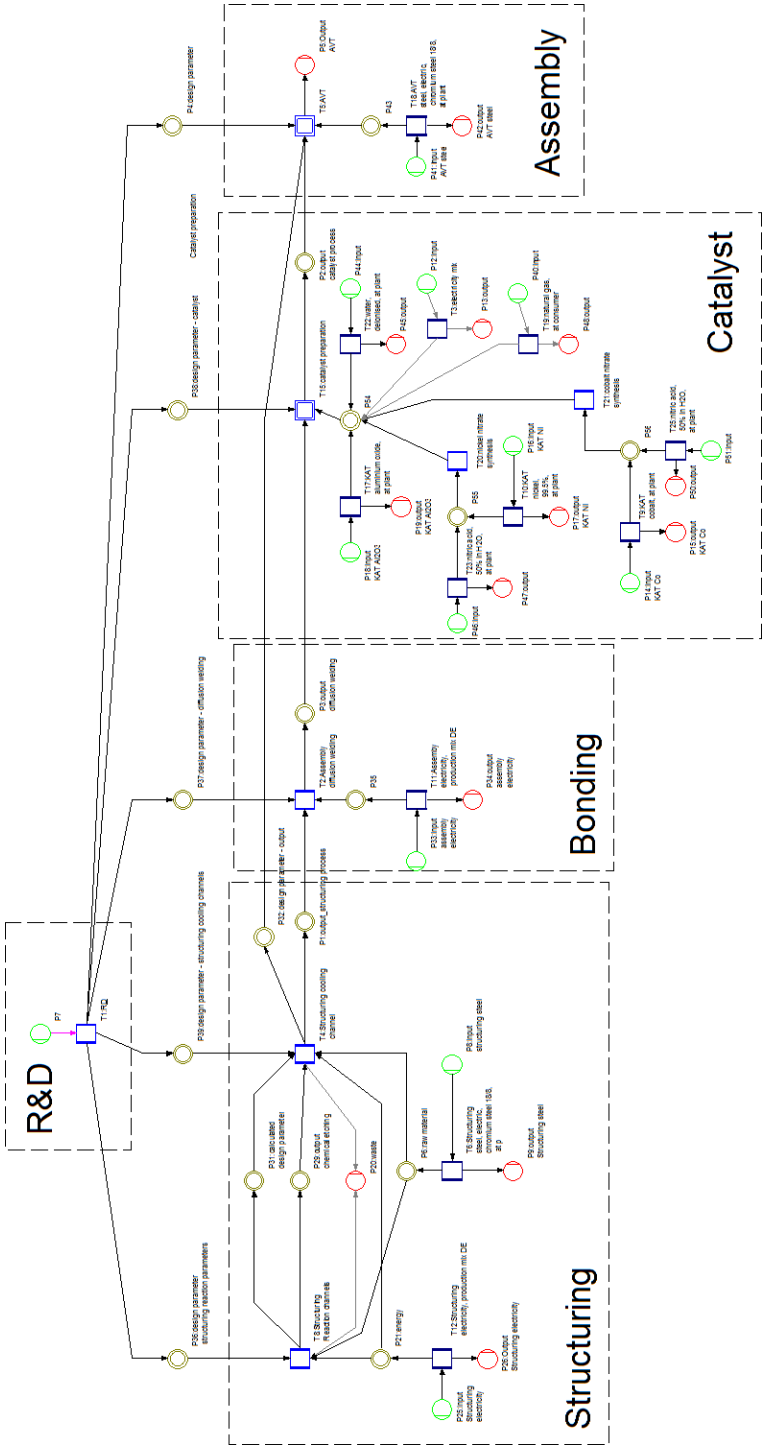

Figure B**.4:** Network of all manufacturing processes using milling for reaction and cooling channel structuring in Umberto®.

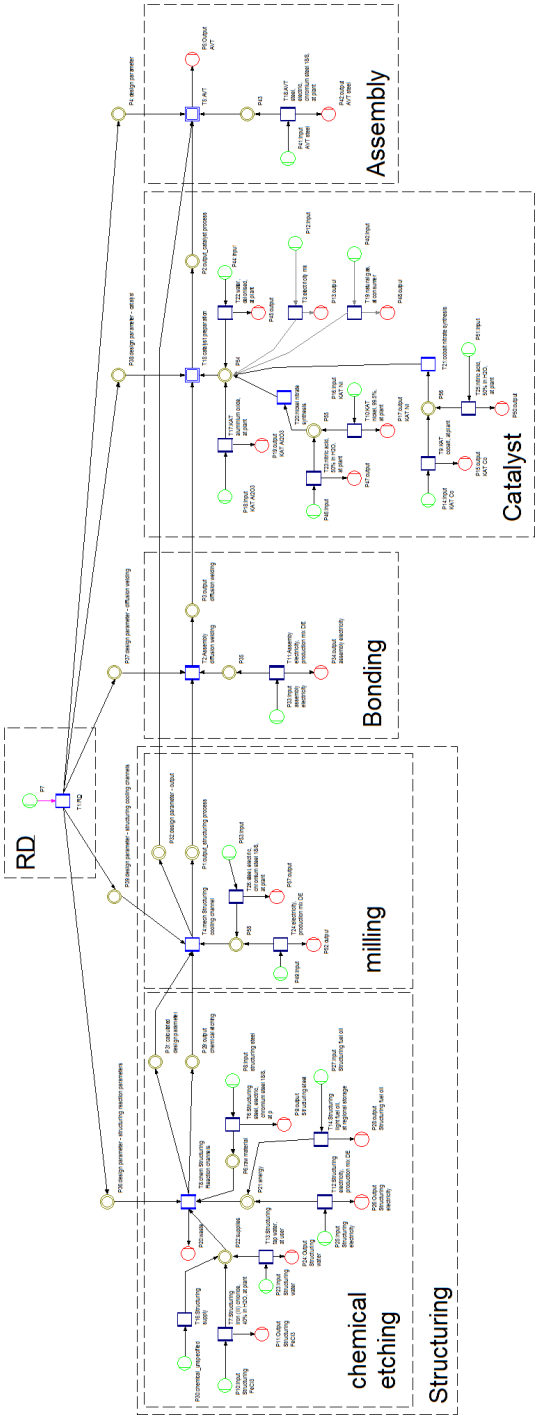

Figure B.**5:** Network of all manufacturing processes using chemical etching for reaction and milling for cooling channel structuring in Umberto®.

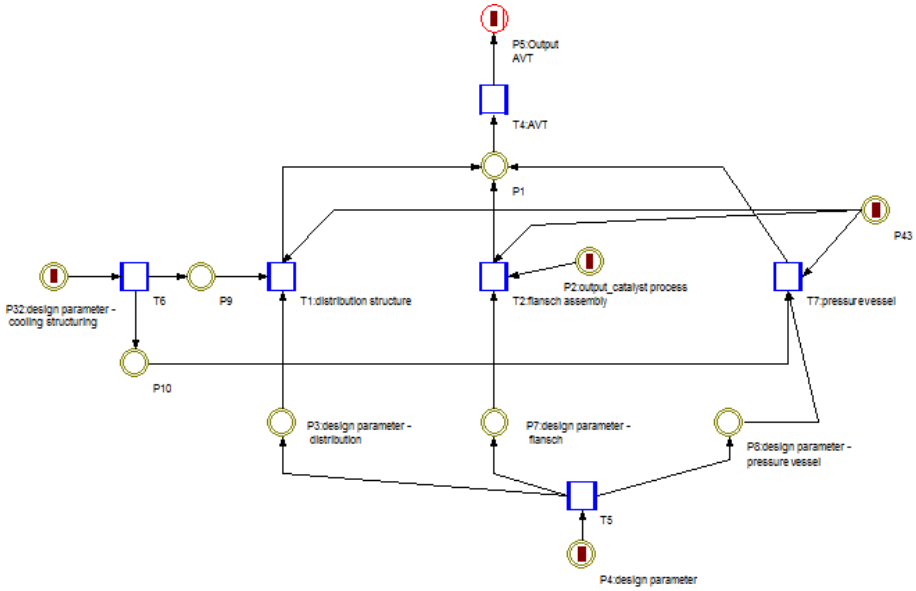

Figure B**.6:** Network of the assembly processes.

3.2 Transition specification using JavaScript

3.2.2 R&D

```
//Transition Specification Functions T1
//Project Modul Mikroreaktor
//Scenario Manufacturing_FTS_ChemStruc
//Created by Eva Zschieschang

if (Tr.CheckAssignment("X00")) {
  output = Tr.Value("X00"); // estimated product (fraction c5+) per hour in kg
  // product distribution ist not included
  Y00 = 0; //catalyst
  Y01 = 0; //V_channel
  Y02 = 0; //wt_Al2O3
  Y03 = 0; //wt_Co
  Y04 = 0; //wt_Ni
  Y05 = 0; //factor_cooling
  Y06 = 0; //factor_distribution
  Y07 = 0; //factor_flansch
  Y08 = 0; //factor_pressure_vessel
  Y09 = 0; //density_raw_material
  Y10 = 0; //factor_welding_loss
  //Y11 = 0; //zuschnitt_hight
  Y12 = 0; //zuschnitt_lenght
  Y13 = 0; //zuschnitt_width

// data: IMVT (presentation FuE commisson; Myrstad, R. et al.)
// Myrstad et al. 2009, Catalyst Typ B, T=225°C, p=20 bar,

   time = 1; // h hour !!!
   productivity = Tr.Value("N00"); // 1.9, in kg_C5+/ kg_cat * h
   bulkDensCat = Tr.Value("N02"); //1000, in kg/m3, unprecise value!
   channelFact = Tr.Value("N03");  // 0.5, part of the channel volume which
    is used for the catalyst

//Output

// Y00 (catalyst)
   catalystMass = output/productivity;  // productivity=N00, weight of catalyst in kg

// Y01 (V_etched)
   catalystVol = catalystMass/bulkDensCat; // in m3
   channelVol= catalystVol/channelFact; //

// parameter setting
   wt_Co = Tr.Value("N10"); // *100, in %
   wt_Ni = Tr.Value("N09"); //*100, in %
   wt_Al2O3 = 1 - (wt_Co + wt_Ni); // *100, in %
   factor_cooling  = Tr.Value("N04"); // 5, reaction channel number : cooling channel number
   factor_distribution  = Tr.Value("N05"); // 1, mass distribution structure =
    mass reaction structure, (20 % distribution volume)
   factor_flansch  = Tr.Value("N06"); //2, mass of the flansch in kg
   factor_pressure_vessel = Tr.Value("N07"); // 0.1, *100, in %; Sicherheitszuschlag
   density_raw_material = 8000; // in kg/m3 of steel
   factor_welding_loss = Tr.Value("N08"); //0.2, willkuerlich gesetzt!!!,
      Energieverbrauch Q beim diffusionscshweißen um 20% höher
   zuschnitt_hight = 0.15/1000000   ;// material proberty in km
   zuschnitt_lenght  = 400/1000000 ; // material proberty in km
   zuschnitt_width  =  300/1000000  ; // material proberty in km

//Save Flows
  Tr.SetValue("Y00",catalystMass);
  Tr.SetValue("Y01",channelVol);
  Tr.SetValue("Y02",wt_Al2O3);
  //Tr.SetValue("Y03",wt_Co);
  //Tr.SetValue("Y04",wt_Ni);
  Tr.SetValue("Y05",factor_cooling);
  Tr.SetValue("Y06",factor_distribution);
  Tr.SetValue("Y07",factor_flansch);
  Tr.SetValue("Y08",factor_pressure_vessel);
  Tr.SetValue("Y09",density_raw_material);
  Tr.SetValue("Y10",factor_welding_loss);
```

```
  //Tr.SetValue("Y11",zuschnitt_hight);
  Tr.SetValue("Y12",zuschnitt_lenght);
  Tr.SetValue("Y13",zuschnitt_width);

} else {
  Tr.AddTempWarning("Some Variables of {X00} unassigned.");
}
```

3.2.2 Reaction Channel
Process - chemical etching

```
//Transition Specification Functions T8 - Structuring Reaction Channel
//Project Modul Mikroreaktor
//Scenario Module Manufacturing_FTS_ChemStruc
//Created by Eva Zschieschang
if (Tr.CheckAssignment("X05")) {
  etchedVol = Tr.Value("X05")//V_channel, design parameter
  X00 = 0 //FeCl3_40
  X01 = 0 //chemicals
  X02 = 0 //electricity
  X03 = 0 //fuel oil
  X04 = 0 //Me_foil (mass raw material)
  X06 = 0 //water
  Y00 = 0 //Me_etched
  Y01 = 0 //no_recycling
  Y02 = 0 //recycling_direct
  Y03 = 0 //recycling_indirect
  Y04 = 0 //sewage
  Y05 = 0 //waste_total
  Y06 = 0 //foil number
  Y07 = 0 //V_etched
  Y08 = 0 //m_zuschnitt_structured
  Y09 = 0 //m_zuschnitt
  Y10 = 0 //V_cooling

// Constatnt declaration

   densityMat = 8000; // kg/m3 , data:
   densityFueloil = 0.86; // l/kg , data: ecoinvent-report No1
   densityWater = 1; // kg/l

// data: IMVT chemical etching
   zuschnittVol = 400*300*0.15/1000000000; // m3
   zuschnittMass = zuschnittVol * densityMat; // kg
   lossCut = 0.2; //  20% material loss through cutting
   lossEtch = 0.44; // 44% material loss through etching (etched volume 0.1*0.1)
   sructuredMass = zuschnittMass * (1-lossCut) * (1-lossEtch); // in kg per zuschnitt
   zuschnittEtchMass = zuschnittMass - sructuredMass; // in kg per zuschnitt
   reactorEtchMass = etchedVol * densityMat; //in kg per microreactor

// data: IMVT, Peter Pfeifer, consumption per zuschnitt
   wastetotal  = 3.11; // kg/zuschnittVol
   recyclingdirect = 0.086; // kg/zuschnittVol
   recyclingindirect = 0.92; // kg/zuschnittVol
   fueloil = 0.29; // l/zuschnittVol
   electricity = 4.94; // kWh/zuschnittVol
   water = 30.142; // l/zuschnittVol
   sewage = 23.085; // l/zuschnittVol

// data: IMVT, Peter Pfeifer, consumption per kg etched material
   FeCl3 = 69.5; // in kg per 1kg etched material, FeCl3 40%ig
   chemicals = 102.25; // in kg per 1 kg etched material

// lossMat calculates structured material per raw material per zuschnitt
// influencing value: lossCut, lossEtch
// non dimensional
   lossMat = (1-lossCut) * (1-lossEtch); //  defines a variable

//Input calculation

// X04 "Me foil" (rawMat) in kg
// reactorEtchMass/ lossEtch = amount of material after zuschnitt loss
// (1-lossCut) = amount of material after zuschnitt loss
   rawMat = (reactorEtchMass/lossEtch)/(1-lossCut);
// calculation: number of necessary "zuschnitt" based on structuredMat
   zuschnittNum = rawMat/zuschnittMass;
// data: IMVT, assumption
// calculation of the necessary cooling channel with the assumption:
// reaction channel volume per zuschnitt = cooling channel volume per zuschnitt
   foil_ratio = 5; // reaction channel/cooling channel
   zuschnittNumCooling = zuschnittNum/foil_ratio;
```

```
   rawMatColling = zuschnittNumCooling * zuschnittMass;
   V_cooling = etchedVol/zuschnittNumCooling;


// X02 "electricity" in kJ, 1kWh = 3600 kJ
  X02 = zuschnittNum*electricity*3600;
// X03 "fuel oil" in kg
  X03 = zuschnittNum*(fueloil*densityFueloil);
// X06 "water" in kg
   X06 = zuschnittNum * water * densityWater;
// X01 "chemicals" in kg
   X01 = zuschnittNum * chemicals;
// X00 "FeCl3_40" in kg
   X00 = zuschnittNum * FeCl3;

//Output
// Y00 "Me_etched" (reactorWeight); microreactor weight
reactorWeight = (reactorEtchMass/lossEtch)- reactorEtchMass;
// Y01 "no recycling"  in kg
Y01 = zuschnittNum * (wastetotal - recyclingdirect - recyclingindirect);
// Y02 "recycling_direct" in kg
Y02 = zuschnittNum * recyclingdirect;
// Y03 "recycling_indirect" in kg
Y03 = zuschnittNum * recyclingindirect;

// Y04 "sewage" in kg
Y04 = zuschnittNum * sewage;

// Y05 "waste total" in kg
Y05 = zuschnittNum * wastetotal;

// Y07 "V_etched" for the reaction in m3
Y07 = reactorWeight/densityMat;

  //Save Flows
  Tr.SetValue("X00",X00)//FeCl3_40
  Tr.SetValue("X01",X01)//chemicals
  Tr.SetValue("X02",X02)//electricity
  Tr.SetValue("X03",X03)//fuel oil
  Tr.SetValue("X04",rawMat)//Me_foil
  Tr.SetValue("X06",X06)//water
  Tr.SetValue("Y00",reactorWeight)//Me_etched
  Tr.SetValue("Y01",Y01)//no_recycling
  Tr.SetValue("Y02",Y02)//recycling_direct
  Tr.SetValue("Y03",Y03)//recycling_indirect
  Tr.SetValue("Y04",Y04)//sewage
  Tr.SetValue("Y05",Y05)//waste_total
  Tr.SetValue("Y06",zuschnittNum)//foil number
  Tr.SetValue("Y07",Y07)//V_etched
  Tr.SetValue("Y08",sructuredMass)//m_zuschnitt_structured
  Tr.SetValue("Y09",zuschnittMass)//m_zuschnitt
  Tr.SetValue("Y10",V_cooling)//Volume for the cooling channel

} else {
  Tr.AddTempWarning("Some Variables of {X05} unassigned.");
  }
```

Process - milling

```
//Transition Specification Functions T1
//Project Modul Mikroreaktor
//Scenario Manufacturing_FTS_MechStruc
//Created by Eva Zschieschang
if (Tr.CheckAssignment("X00")) {
  etchedVol = Tr.Value("X00")//V_channel, design parameter, = milledVol
  X01 = 0 //electricity
  X02 = 0 //Me_foil (mass raw material, steel)
  Y00 = 0 //Me_etched
  Y01 = 0 //foil number
  Y02 = 0 //V_etched
  Y03 = 0 //m_zuschnitt_structured
  Y04 = 0 //m_zuschnitt
  Y05 = 0 //V_cooling

// Constatnt declaration
   densityMat = 8000; // kg/m3 , data:

// data: IMVT mechanical milling (material loss equivalent to chemical etching, has not be proven with IMVT)
// the term etching is here used for milling
   zuschnittVol = 400*300*0.15/1000000000; // m3
   zuschnittMass = zuschnittVol * densityMat; // kg
   lossCut = 0.2; //  20% material loss through cutting
   lossEtch = 0.44; // 44% material loss through etching (etched volume 0.1*0.1)
   structuredMass = zuschnittMass * (1-lossCut) * (1-lossEtch); // in kg per zuschnitt, = part of the reactor mass
   zuschnittEtchMass = zuschnittMass - structuredMass; // in kg per zuschnitt
   reactorEtchMass = etchedVol * densityMat; //in kg per microreactor

// data: IMVT, Lutz Eichhornconsumption., Zerspannung in Edelstahl 1.4301
// Bearbeitungsparameter für Vollhartmetallschaftfräser, zylindrisch Fa. Hitachi, Japan
   Schnitttiefe =  0.1; // in mm
   Drehzahl = 19000; // in mm/min
   Vorschub = 0.040; // in mm
   SchnittGeschw = 1520; // in mm/min
   Fraeslaenge = 180; // in m, max Länge für einen Fräskopf, danach muss dieser wg. Verschleis ausgewechselt werden
   Wirkungsgrad = 0.8; // dimensionslos, fiktive Annahme
   FaktorVerschleiss = 20; //Maschine wird um den Verschleißfaktor größer dimensioniert
   FaktorZeit   = 0.1; // 10 % mehr Zeit als berechnet

// Werkstoffeigenschaften für Edelstahl 1.4301
   kc11 = 3811; // in N/mm2, spezifische Schnittkraft
   Anstiegswert = 0.75; //dimensionslos, 1-z, für Werkstoff Ck 45 V (Vergütungsstahl)
   Einstellwinkel = 60; // in °, fiktiv gewählt aus Literatur

// lossMat calculates structured material per raw material per zuschnitt
// influencing value: lossCut, lossEtch
// non dimensional
   lossMat = (1-lossCut) * (1-lossEtch); //  defines a variable

//Input calculation

// X02 "Me foil" (rawMat) in kg
// reactorEtchMass/ lossEtch = amount of material after zuschnitt loss
// (1-lossCut) = amount of material after zuschnitt loss
   rawMat = (reactorEtchMass/lossEtch)/(1-lossCut);

// calculation: number of necessary "zuschnitt" based on structuredMat
   zuschnittNum = rawMat/zuschnittMass;

// data: IMVT, assumption
// calculation of the necessary cooling channel with the assumption:
// reaction channel volume per zuschnitt = cooling channel volume per zuschnitt
   foil_ratio = 5; // reaction channel/cooling channel
   zuschnittNumCooling = zuschnittNum/foil_ratio;
   rawMatColling = zuschnittNumCooling * zuschnittMass;
   V_cooling = etchedVol/zuschnittNumCooling;

// X01 "electricity" in kJ, 1kWh = 3600 kJ
// calculation with the Kienzle-Gleichung
// Fc = kc1.1 * b * h^(1-z)
   sin60 =  0.8660;
   b = Schnitttiefe/sin60; // Spanungsbreite in mm
```

```
   h = Vorschub*sin60; //Spanungsdicke in mm
// Fc = kc11 * b * wurzel_h;// in N;
   Fc = 35; // in N, Wurzelfunktion doesnt work

// calculation of the necessary energy
// P = Fc*Schnittgeschw/Wirkungsgrad in W
   P = (Fc * (SchnittGeschw/(1000*60)))/Wirkungsgrad;//N*m/s =Ws/s = W
   PM = P*FaktorVerschleiss; //in W
   length_channel =  etchedVol/(0.0001*0.0001); // Kanalbreite*Kanaltiefe in mm & umgerechnet in m, Parameter von IMVT
   number_milling = length_channel/Fraeslaenge; // Anzahl an Fräsen!!! - sehr hohe Zahl
   time =  length_channel/(SchnittGeschw/(1000*60)); // time for channel milling in s, Faktor 60*1000 für Umrechnung mm/min in m/s
   energy = (PM*(time/3600))/1000; // in  kWh

//Output
// Y00 "Me_etched" (reactorWeight); microreactor weight
reactorWeight = (reactorEtchMass/lossEtch)- reactorEtchMass;
// Y02 "V_etched" for the reaction in m3
Y02 = reactorWeight/densityMat;

//Save Flows
  Tr.SetValue("X01",energy)//electricity
  Tr.SetValue("X02",rawMat)//Me_foil, raw Material

  Tr.SetValue("Y00",reactorWeight)//Me_etched
  Tr.SetValue("Y01",zuschnittNum)//foil number
  Tr.SetValue("Y02",Y02)//V_etched
  Tr.SetValue("Y03",structuredMass)//m_zuschnitt_structured
  Tr.SetValue("Y04",zuschnittMass)//m_zuschnitt
  Tr.SetValue("Y05",V_cooling)//Volume for the cooling channel
} else {
  Tr.AddTempWarning("Some Variables of {X00} unassigned.");
}
```

3.2.2 Cooling Channel

Process - chemical etching

```
//Transition Specification Functions T4 - Structuring Cooling Channel
//Project Modul Mikroreaktor
//Scenario Module Manufacturing 3
//Created by Eva Zschieschang

if (Tr.CheckAssignment("X00,X07,X08,X09,X10,X11,X12")) {
  X00 = Tr.Value("X00");//Me_etched
  foil_number = Tr.Value("X07");// foil_number
  X08 = Tr.Value("X08");//V_cooling
  etchedVol = Tr.Value("X09");//V_etched - reaction channel
  X10 = Tr.Value("X10");//m_zuschnitt_structured
  m_zuschnitt = Tr.Value("X11");//m_zuschhnitt
  foil_ratio = Tr.Value ("X12"); //factor cooling, foil number_reaction channel
  X01 = 0;//electricity
  X02 = 0;//fuel oil
  X03 = 0;//water
  X04 = 0;//chemicals
  X05 = 0;//steel
  X06 = 0;//FeCl3
  Y00 = 0;//m_channel_cooling
  Y01 = 0;//no_recycling
  Y02 = 0;//recycling_direct
  Y03 = 0;//recycling_indirect
  Y04 = 0;//sewage
  Y05 = 0;//waste total
  Y06 = 0;//Me_etched
  Y07 = 0;//reactor hight

// Constatnt declaration
   densityMat = 8000; // kg/m3 , data:
   densityFueloil = 0.86; // l/kg , data: ecoinvent-report No1
   densityWater = 1; // kg/l

// data: IMVT chemical etching
   zuschnittVol = 400*300*0.15/1000000000; // m3
   zuschnittMass = zuschnittVol * densityMat; // kg
   lossCut = 0.2; //  20% material loss through cutting
   lossEtch = 0.44; // 44% material loss through etching (etched volume 0.1*0.1)
   sructuredMass = zuschnittMass * (1-lossCut) * (1-lossEtch); // in kg per zuschnitt
   zuschnittEtchMass = zuschnittMass - sructuredMass; // in kg per zuschnitt
   reactorEtchMass = etchedVol * densityMat; //in kg per microreactor

// data: IMVT, Peter Pfeifer, consumption per zuschnitt
   wastetotal  = 3.11; // kg/zuschnittVol
   recyclingdirect = 0.086; // kg/zuschnittVol
   recyclingindirect = 0.92; // kg/zuschnittVol
   fueloil = 0.29; // l/zuschnittVol
   electricity = 4.94; // kWh/zuschnittVol
   water = 30.142; // l/zuschnittVol
   sewage = 23.085; // l/zuschnittVol

// data: IMVT, Peter Pfeifer, consumption per kg etched material
   FeCl3 = 69.5; // in kg per 1kg etched material, FeCl3 40%ig
   chemicals = 102.25; // in kg per 1 kg etched material

// lossMat calculates structured material per raw material per zuschnitt
// influencing value: lossCut, lossEtch
// non dimensional
   lossMat = (1-lossCut) * (1-lossEtch); //  defines a variable

//Input calculation  fot the cooling channel!!!

// X04 "Me foil" (rawMat) in kg
// reactorEtchMass/ lossEtch = amount of material after zuschnitt loss
// (1-lossCut) = amount of material after zuschnitt loss
   rawMat = (reactorEtchMass/lossEtch)/(1-lossCut);

// calculation: number of necessary "zuschnitt" based on structuredMat
   zuschnittNum = rawMat/zuschnittMass;
```

```
// data: IMVT, assumption
// calculation of the necessary cooling channel with the assumption:
// reaction channel volume per zuschnitt = cooling channel volume per zuschnitt

   zuschnittNumCooling = foil_number/foil_ratio;
   rawMatColling = zuschnittNumCooling * zuschnittMass;
   V_cooling = etchedVol/zuschnittNumCooling;

  // calculation mass
  // including cooling structure
   m_cooling =  zuschnittNumCooling *  m_zuschnitt;

  //including reaction and cooling structures  = Y00
  // in kg; assumption: mass of structured cooling foils = mass of structured reaction foils
  //X00 = Me_etched (Weigth of all structured foils), X10 = m_zuschnitt_structured
   (weigth of 1 structured Zuschnitt
   m_channel_cooling = X00 + (zuschnittNumCooling*X10);

// X01 "electricity" in kJ, 1kWh = 3600 kJ
   X01 = zuschnittNumCooling*electricity*3600;
// X02 "fuel oil" in kg
   X02 = zuschnittNumCooling*(fueloil*densityFueloil);
// X03 "water" in kg
   X03 = zuschnittNumCooling * water * densityWater;
// X04 "chemicals" in kg
   X04 = zuschnittNumCooling * chemicals;
// X06 "FeCl3_40" in kg
   X06 = zuschnittNumCooling * FeCl3;

//Output
// Y01 "no recycling"  in kg
   Y01 = zuschnittNumCooling * (wastetotal - recyclingdirect - recyclingindirect);
// Y02 "recycling_direct" in kg
   Y02 = zuschnittNumCooling * recyclingdirect;
// Y03 "recycling_indirect" in kg
   Y03 = zuschnittNumCooling * recyclingindirect;
// Y04 "sewage" in kg
   Y04 = zuschnittNumCooling * sewage;
// Y05 "waste total" in kg
   Y05 = zuschnittNumCooling * wastetotal;
// Y07 "reactor hight" in km
   Y07 = ((zuschnittNumCooling + foil_number)* 0.15)/1000000;

  //Save Flows
  Tr.SetValue("X01",X01);//electricity
  Tr.SetValue("X02",X02);//fuel oil
  Tr.SetValue("X03",X03);//water
  Tr.SetValue("X04",X04);//chemicals
  Tr.SetValue("X05",rawMatColling);//steel
  Tr.SetValue("X06",X06);//FeCl3
  Tr.SetValue("Y00",m_channel_cooling);//m_channel_cooling
  Tr.SetValue("Y01",Y01);//no_recycling
  Tr.SetValue("Y02",Y02);//recycling_direct
  Tr.SetValue("Y03",Y03);//recycling_indirect
  Tr.SetValue("Y04",Y04);//sewage
  Tr.SetValue("Y05",Y05);//waste total
  Tr.SetValue("Y06",X00);//Me_etched
  Tr.SetValue("Y07",Y07);//reactor hight
} else {
  Tr.AddTempWarning("Some Variables of {X00,X07,X08,X09,X10,X11} unassigned.");
}
```

Process - milling

```
//Transition Specification Functions T2
//Project Modul Mikroreaktor
//Scenario Module Manufacturing_FTS_MechStruc
//Created by Eva Zschieschang

if (Tr.CheckAssignment("X00,X03,X04,X05,X06,X07,X08")) {
  Me_etched = Tr.Value("X00");//Me_etched, reactor weigth reaction channels
  foil_number = Tr.Value("X03");// foil_number
  X04 = Tr.Value("X04");//V_cooling
  etchedVol = Tr.Value("X05");//V_etched, = microreactor volume
  m_zuschnitt_structured = Tr.Value("X06");//m_zuschnitt_structured
  m_zuschnitt = Tr.Value("X07");//m_zuschhnitt
  foil_ratio = Tr.Value ("X08"); //factor cooling, foil number_reaction channel
  X01 = 0;//electricity
  X02 = 0;//steel
  Y00 = 0;//m_channel_cooling
  Y01 = 0;//Me_etched
  Y02 = 0;//reactor hight

// Constatnt definition
   densityMat = 8000; // kg/m3 , data:

// data: IMVT mechanical milling (material loss equivalent to chemical etching, has not be proven with IMVT)
// the term etching is here used for milling
   zuschnittVol = 400*300*0.15/1000000000; // m3
   zuschnittMass = zuschnittVol * densityMat; // kg
   lossCut = 0.2; //  20% material loss through cutting
   lossEtch = 0.44; // 44% material loss through etching (etched volume 0.1*0.1)
   structuredMass = zuschnittMass * (1-lossCut) * (1-lossEtch); // in kg per zuschnitt
   zuschnittEtchMass = zuschnittMass - structuredMass; // in kg per zuschnitt
   reactorEtchMass = etchedVol * densityMat; //in kg per microreactor with reaction channels

// data: IMVT, Lutz Eichhornconsumption., Zerspannung in Edelstahl 1.4301
// Bearbeitungsparameter für Vollhartmetallschaftfräser, zylindrisch Fa. Hitachi, Japan
   Schnitttiefe =  0.1; // in mm
   Drehzahl = 19000; // in mm/min
   Vorschub = 0.040; // in mm
   SchnittGeschw = 1520; // in mm/min
   Fraeslaenge = 180; // in m, max Länge für einen Fräskopf, danach muss dieser wg. Verschleis ausgewechselt werden
   Wirkungsgrad = 0.8; // dimensionslos, fiktive Annahme
   FaktorVerschleiss = 2; //Maschine wird um den Verschleißfaktor größer dimensioniert
   FaktorZeit   = 0.1; // 10 % mehr Zeit als berechnet

// Werkstoffeigenschaften für Edelstahl 1.4301
   kc11 = 3811; // in N/mm2, spezifische Schnittkraft
   Anstiegswert = 0.75; //dimensionslos, 1-z, für Werkstoff Ck 45 V (Vergütungsstahl)
   Einstellwinkel = 60; // in °, fiktiv gewählt aus Literatur

// lossMat calculates structured material per raw material per zuschnitt
// influencing value: lossCut, lossEtch
// non dimensional
   lossMat = (1-lossCut) * (1-lossEtch); //  defines a variable

// Input calculation
// X02 "Me foil" (rawMat) in kg
// reactorEtchMass/ lossEtch = amount of material after zuschnitt loss
// (1-lossCut) = amount of material after zuschnitt loss
   rawMat = (reactorEtchMass/(1-lossEtch))/(1-lossCut);

// calculation: number of necessary "zuschnitt" based on structuredMat
   zuschnittNum = rawMat/zuschnittMass;

// data: IMVT, assumption
// calculation of the necessary cooling channel with the assumption:
// reaction channel volume per zuschnitt = cooling channel volume per zuschnitt
   foil_ratio = 5; // reaction channel/cooling channel
   zuschnittNumCooling = zuschnittNum/foil_ratio;
   rawMatColling = zuschnittNumCooling * zuschnittMass;
// V_cooling = etchedVol/zuschnittNumCooling;
// X01 "electricity" in kJ, 1kWh = 3600 kJ
// calculation with the Kienzle-Gleichung
// Fc = kc1.1 * b * h^(1-z)
```

```
   sin60 =  0.8660;
   b = Schnitttiefe/sin60; // Spanungsbreite in mm
   h = Vorschub/sin60; //Spanungsdicke in mm
//Fc = kc11 * b * wurzel(Anstiegswert,h); // in N
   Fc = 35; // in N

// calculation of the necessary energy
// P = Fc*Schnittgeschw/Wirkungsgrad in W
   P = (Fc * (SchnittGeschw/(1000*60)))/Wirkungsgrad;//N*m/s =Ws/s = W
   PM = P*FaktorVerschleiss; //in W
   length_channel = ((etchedVol/56)*44)/(0.0001*0.0001); // Kanalbreite*Kanaltiefe in mm & umgerechnet in m, Parameter von IMVT
   number_milling = length_channel/Fraeslaenge; // Anzahl an Fräsen!!! - sehr hohe Zahl
   time =  length_channel/(SchnittGeschw/(1000*60)); // time for channel milling in s, Faktor 60*1000 für Umrechnung mm/min in m/s
   energy = (PM*(time/3600))/1000; // in  kWh

//Input calculation  for the cooling channel!!!
// = 1/fiol ration of energy consumption, because there is 1/foil ration cooling channel length
   energy_cooling = energy/foil_ratio;

// data: IMVT, assumption
// calculation of the necessary cooling channel with the assumption:
// reaction channel volume per zuschnitt = cooling channel volume per zuschnitt
   zuschnittNumCooling = foil_number/foil_ratio;
   rawMatColling = zuschnittNumCooling * zuschnittMass;

// calculation mass
// including cooling structure
   m_cooling =  zuschnittNumCooling *  m_zuschnitt;

//including reaction and cooling structures  = Y00
// in kg; assumption: mass of structured cooling foils = mass of structured reaction foils
   m_channel_cooling = Me_etched + (zuschnittNumCooling*m_zuschnitt_structured);
// Y02 "reactor hight" in km = rector_size
   Y02 = ((zuschnittNumCooling + foil_number)* 0.15)/1000000;

//Save Flows
  Tr.SetValue("X01",energy_cooling);//electricity
  Tr.SetValue("X02",rawMatColling);//steel
  Tr.SetValue("Y00",m_channel_cooling);//m_channel_cooling, mass reaction and cooling channel
  Tr.SetValue("Y01",Me_etched);// Me_etched
  Tr.SetValue("Y02",Y02); //reactor hight = reactor size
} else {
  Tr.AddTempWarning("Some Variables of {X00,X07,X08,X09,X10,X11} unassigned.");
}
```

3.2.2 Bonding

```
//Transition Specification Functions T2 - Diffusion Welding
//Project Modul Mikroreaktor
//Scenario Module Manufacturing_FTS_ChemStruc
//Created by Eva Zschieschang

if (Tr.CheckAssignment("X00,X02")) {
  m_channel_cooling = Tr.Value("X00");//m_channel_cooling
  X02 = Tr.Value("X02");//factor_welding_loss (0.x in %)
  X01 = 0; //electricity
  Y00 = 0; //m_channel_cooling

  // parameter definition
  // data source: http://www.schweizer-fn.de/stoff/wkapazitaet/v2_wkapazitaet_metall.htm
  //data source: IMVT
  c_p = 477;  // in J/kg*K
  T = 1000;   // temperature difference in K for the heating of the steel;
  //x_loss = 0.3 // efficiency loss of the machine, efficiency = 1-x_loss
  foil_size = 0.15 // in mm; foil hight, data source: IMVT

  // calculation of the needed heat flow volume
  Q = (m_channel_cooling*c_p*T)/3600000;  // in kWh

  // electricity calculation
  X01 = Q * (1+ X02);

  //Save Flows
  Tr.SetValue("X01",X01) //electricity
  Tr.SetValue("Y00",m_channel_cooling)//mass of the reactor including
   reaction channels and distribution structure

} else {
  Tr.AddTempWarning("Some Variables of {X00,X02} unassigned.");
}
```

3.2.2 Catalyst

```
//Transition Specification Functions T1
//Project Modul Mikroreaktor
//Scenario Modul Manufacturing_FTS_ChemStruc
//Created by Eva Zschieschang

if (Tr.CheckAssignment("X04,X05,X06")) {
  X04 = Tr.Value("X04");  //m_channel_cooling
  X05 = Tr.Value("X05");  //m_catalyst
  X06 = Tr.Value("X06");  //wt_Al2O3

  X00 = 0; //Co
  X01 = 0; //Ni
  X02 = 0; //Al2O3
  X03 = 0; // water_deionised
  Y00 = 0;
  Y01 = 0;

  //import network parameters
  wt_Co = Tr.Value("N10");
  wt_Ni = Tr.Value("N09");
  wt_Al2O3 = X06;
  m_catalyst = X05;

  X00 = m_catalyst * wt_Co;
  X01 = m_catalyst * wt_Ni;
  X02 = m_catalyst * wt_Al2O3;
  X03 = m_catalyst *10; // diluted 1:10, calculate the amount of water

  //Save Flows
  Tr.SetValue("X00",X00);
  Tr.SetValue("X01",X01);
  Tr.SetValue("X02",X02);
  Tr.SetValue("X03",X03);
  Tr.SetValue("Y00",m_catalyst); //m_catalyst
  Tr.SetValue("Y01",X04);
} else {
  Tr.AddTempWarning("Some Variables of {Y00,Y01} unassigned.");
}
```

3.2.2 Pressure Vessel

```
//Transition Specification Functions T1_pressure vessel
//Project Modul Mikroreaktor
//Scenario Manufacturing_FTS_ChemStruc
//Created by Eva Zschieschang

function sqr(x)
{
  var y=x*x;
  return y;
}

//Math.sqrt()    function to calculate the square root

if (Tr.CheckAssignment("X01,X04,X05,X06,X07")) {
  reactor_hight = Tr.Value("X01"); //reactor hight = reactor size   in km
  factor_pressure_vessel = Tr.Value("X04");
  zuschnitt_lenght = Tr.Value("X05"); // in km
  zuschnitt_width = Tr.Value("X06");  // in km
  density_raw_material = Tr.Value("X07") // in kg/m3
  X03 = 0;
  Y00 = 0;

//calculation of the presure vessel mass using the "Kesselformel, Bockwurstformel"

//data source: Strackeljan /Festigkeit, Grundlagen der Festigkeitslehre
  sigma = 230; // in N/mm2, zulässige Normalspannung, hier Dehngrenze 1.4301, EN 10088-2,
    Bemessungshilfen zu nichtrostende Stählen im Bauwesen
  delta_p = Tr.Value("N11"); // pressure in bar, 1bar = 0.1 N/mm2, data source: IMVT

//calculation of the reactor  diameter
  diameter_sqr = sqr(zuschnitt_lenght)+ sqr(zuschnitt_width);//sqr(zuschnitt_width) was missing,
   added at 14.05.2012
  diameter_reactor = Math.sqrt(diameter_sqr); //14.05.2012 correct function of Math.sqrt() inserted
  radius_reactor =  diameter_reactor/2; // in km

//calculation of the volume of the pressure vessel
  diameter_vessel = diameter_reactor * (1 + factor_pressure_vessel); // in km
  radius_vessel = diameter_vessel/2; //in km
  vessel_hight = reactor_hight * (1 + factor_pressure_vessel)*1000; // in m
  pi = 3.14159265; //Pi
  V_vessel = pi*sqr(radius_vessel)* 1000000 * vessel_hight; //in m3

//calculation of the wall thickness
//Kesselformel: sigma = p*d/2*s (p - Druck, d - Durchmesser, s - Wandstärke)
  s = ((delta_p*0.1) *(diameter_vessel*1000000))/(2*sigma); // in mm

//unit transformation
  diameter_vessel_min = diameter_vessel*1000;// im m
  diameter_vessel_max = (diameter_vessel*1000)+ (s/1000);// includes the wallthickness s, im m

// calculation of the vessel mass
  A_vessel_min =  (sqr(diameter_vessel_min)*pi)/4; // in m2
  A_vessel_max =  (sqr(diameter_vessel_max)*pi)/4; // in m2

  V_vessel_cleading = (A_vessel_max - A_vessel_min) * vessel_hight;  // in m3
  m_panel = 2 *( A_vessel_max * (s/1000 )* density_raw_material); //density raw material in kg/m3,
   s in mm, upper and lower side of the pressure vessel, in kg
  m_pressure_vessel = V_vessel_cleading * density_raw_material + m_panel;
   //density raw material in kg/m3, in kg

//Save Flows
  Tr.SetValue("X03",m_pressure_vessel); //input steel
  Tr.SetValue("Y00",m_pressure_vessel); // m_vessel
  Tr.SetValue("Y01",s); //VesselWallThickness in mm
  Tr.SetValue("Y02",diameter_vessel);//VesselDiameter in km
  Tr.SetValue("Y03",V_vessel_cleading); //VesselMantel in m3
  Tr.SetValue("Y04",diameter_sqr);
  Tr.SetValue("Y05",diameter_reactor);  //diameter_reactor
  Tr.SetValue("Y06",vessel_hight); //in m
  Tr.SetValue("Y07",V_vessel); // in m3
} else {
  Tr.AddTempWarning("Some Variables of {X00,X01,X02} unassigned.");
```

APPENDIX C

Fischer-Tropsch synthesis

This appendix presents the Matlab source code for the reactant and product calculation for the FTS.

1 Reactant and product calculation in Matlab

```
% Defining molar weight
M_C = 12.001;           % in g/mol
M_H = 1.00794;          % in g/mol
M_H2 = 2*M_H;           % in g/mol
M_O = 15.9994;          % in g/mol
M_CO = (M_C + M_O);     % in g/mol
M_H2O = (2*M_H + M_O);  % in g/mol

%M_C1 = 0;      % in %/ 100 wt.; C1
%M_C2 = 0;      % in %/ 100 wt.; C2
%M_C3 = 0;      % in %/ 100 wt.; C3
%M_C4 = 0;      % in %/ 100 wt.; C4
%M_C5 = 0;      % in %/ 100 wt.; C5+

% Defining considered product composition
% Calculating molar weight of the considered products C1, C2, C3, C4 and C5+ (Alkane)

x1 = 1;   %
M_C1 = (x1 * M_C) + ((2*x1+2)*M_H) ;
x2 = 2;   %
M_C2 = (x2 * M_C) + ((2*x2+2)*M_H) ;
x3 = 3;     %
M_C3 = (x3 * M_C) + ((2*x3+2)*M_H) ;
x4 = 4;     %
M_C4 = (x4 * M_C) + ((2*x4+2)*M_H) ;
x5 = 22.5;     %
M_C5 = (x5 * M_C) + ((2*x5+2)*M_H) ;

%Calculating the chain growth probability
% chain growth probability a is calculated separatly via zero calculation in Matlab for different c5+ selectivities
a = 0.8314;   % S = 0.80
%a = 0.8526  % S = 0.84
%a = 0.8526 % S = 0.88

% Calculatiing the product distribution in % wt.  for each fraction (C1, C2, C3, C4 and C5+)
% basic: Anderson Flory Kinetic Mn = realpow(a,x)*(1-a)
% calculation is done separately in Excel

%Calculating the product weigth is done separatly in Excel via selectivity
%C5+ output = 50 kg/h
```

```
% C1-5+ product distribution
% a = 0.8314
m_C1 = 1776.6;  % in g
m_C2 = 2954.1;  % in g
m_C3 = 3684.1;  % in g
m_C4 = 4083.9;  % in g
m_C5 = 50000;   % in g

%a = 0.8526
%m_C1 = 1293.1;  % in g
%m_C2 = 2205.3;  % in g
%m_C3 = 2820.4;  % in g
%m_C4 = 3206.2;  % in g
%m_C5 = 50000;   % in g

%a = 0.8754
%m_C1 = 882.2;  % in g
%m_C2 = 1544.5;  % in g
%m_C3 = 2028.1;  % in g
%m_C4 = 2367.2;  % in g
%m_C5 = 50000;   % in g

% Calculating the amount of substance of the products

n_C1 = m_C1/M_C1;   % in mol
n_C2 = m_C2/M_C2; % in mol
n_C3 = m_C3/M_C3; % in mol
n_C4 = m_C4/M_C4; % in mol
n_C5 = m_C5/M_C5; % in mol

% Calculating the amount of reactantst
% basic: molar weight M(X) = m/n(X)
% main FTS reaction: nCO + 2n H2 --> n (CH2)n + nH2O
% C1
m_CO_C1 = n_C1 * M_CO/1000;      % educt: carbon monoxid nCO; in kg
m_H2_C1 = 3 * n_C1 *M_H2/1000;    % educt: hydrogen (2n+1) H2; in kg
m_H2O_C1=  n_C1 * M_H2O/1000;    % product: water nH2O; in kg
% C2
m_CO_C2 = 2* n_C2 * M_CO/1000;      % educt: carbon monoxid nCO; in kg
m_H2_C2 = 5 * n_C2 *M_H2/1000;    % educt: hydrogen (2n+1) H2; in kg
m_H2O_C2=  2 * n_C2 * M_H2O/1000;    % product: water nH2O; in kg
% C3
m_CO_C3 = 3 * n_C3 * M_CO/1000;      % educt: carbon monoxid nCO; in kg
m_H2_C3 = 7 * n_C3 *M_H2/1000;    % educt: hydrogen (2n+1) H2; in kg
m_H2O_C3=  3 * n_C3 * M_H2O/1000;    % product: water nH2O; in kg
% C1
m_CO_C4 = 4 * n_C4 * M_CO/1000;      % educt: carbon monoxid nCO; in kg
m_H2_C4 = 9 * n_C4 *M_H2/1000;    % educt: hydrogen (2n+1) H2; in kg
m_H2O_C4=  4 * n_C4 * M_H2O/1000;    % product: water nH2O; in kg
% C1
m_CO_C5 = 22.5 * n_C5 * M_CO/1000;      % educt: carbon monoxid nCO; in kg
m_H2_C5 = 46 * n_C5 *M_H2/1000;    % educt: hydrogen (2n+1) H2; in kg
m_H2O_C5=  22.5 * n_C5 * M_H2O/1000;    % product: water nH2O; in kg

m_CO = m_CO_C1 + m_CO_C2 + m_CO_C3 + m_CO_C4 + m_CO_C5;
m_H2 = m_H2_C1 + m_H2_C2 + m_H2_C3 + m_H2_C4 + m_H2_C5;
m_H2O = m_H2O_C1 + m_H2O_C2 + m_H2O_C3 + m_H2O_C4 + m_H2O_C5;
```

2 Alpha calculation by C_{5+} selectivity

1. Equation formulation

$$W_i = i(1-\alpha)^2\alpha^{i-1} \tag{C.1}$$

$$\sum_{i=1}^{4} W_i = 1 - W_{C_{5+}} \tag{C.2}$$

$$\sum_{i=1}^{4} W_i = -4\alpha^5 + 5\alpha^4 + 1 \tag{C.3}$$

2. Zero calculation in Matlab:

```
px = [-4 5 0 0 0 + Wc5+];
nullst = roots (px)
```

3 Calculation of the weight and the product distribution using Excel

Equations C.4 to C.8 are used for the product distribution (W) and weight (m) calculation in Excel. Results for design B, B1 and B2 are illustrated in Table C.1.

$$S = \frac{m_k}{m_i} \tag{C.4}$$

$$W_{C5+} = 1 - (W_4 + W_3 + W_2 + W_1) \tag{C.5}$$

$$W_i = i * (1-\alpha)^2 * \alpha^{i-1} \tag{C.6}$$

$$m_{total} = \frac{m_{C5+}}{W_{C5+}} \tag{C.7}$$

$$m_{C_i} = m_{total} * W_i \tag{C.8}$$

	Design B1	Design B	Design B2
productivity in $g_{C_{5+}} g_{cat}^{-1} * h^{-1}$	1.9	1.9	1.9
m_{C5+} output in kg h^{-1}	50	50	50
selectivity W_{5+} in %	80	84	88
chain growth probability α	0.8314	0.8526	0.8754
W_{5+} in % wt	0.8000	0.8400	0.8799
W_4 in % wt	0.0653	0.0539	0.0417
W_3 in % wt	0.0589	0.0474	0.0357
W_2 in % wt	0.0473	0.0370	0.0272
W_1 in % wt	0.0284	0.0217	0.0155
m_{C5+} in kg	50	50	50
m_{C4} in kg	4.0839	3.2062	2.3672
m_{C3} in kg	3.6841	2.8204	2.0281
m_{C2} in kg	2.9541	2.2053	1.5445
m_{C1} in kg	1.7766	1.2933	0.8822
m_{total}	62.4987	59.5252	56.8219

Table C.1: Calculated product distribution and weight of the FTS products.

APPENDIX D

Gas-to-Liquid

This appendix presents the Aspen®Plus model implemented by Manuel Selinsek and scenario parameters of the GtL process.

1 Aspen model

1.1 GtL simulation setup

Figure D.1 illustrates the process flow sheet of the GtL process containing reforming (REFORMER), methane oxidation (TOT-OX), FTS (FTR1/2), heat exchange (WT1/2), water separation (H2OTR1/2, FTRKOND1/2), phase separation (SPLIT) and compressor (PUMP1). The simulation is divided into two operation steps; first without recycling material and heat flows (start of the GtL process) and second with recycling (run the GtL process).

In the first step, natural gas (input flow GAS) is compressed to the defined system pressure in COMP1, pre-heated to 500 °C in HEATER1 and then fed to the REFORMER. Water as input flow H2OEIN in steam reforming is brought to system pressure in liquid stage in PUMP1, heated and evaporated at 800 °C in the heat exchanger WT1 and then fed to the REFORMER. Afterwards, natural gas and steam are converted to synthesis gas at 900 °C in the REFORMER. The 900 °C hot product flow consisting of synthesis gas, steam and not-converted natural gas is used for the heating and evaporation of the water in the heat exchanger WT1. The steam is separated from the product flow through condensation at 50 °C in H2OTR1 and then removed from the process (output flow H2OAUS). Subsequent to reforming the product flow consisting of synthesis and natural gas - mainly methane - is heated to the necessary temperature of the FTS process in HEATER2 and then fed to the FTS reactor FTR1. Water and alkane with longer C-chains as side products of

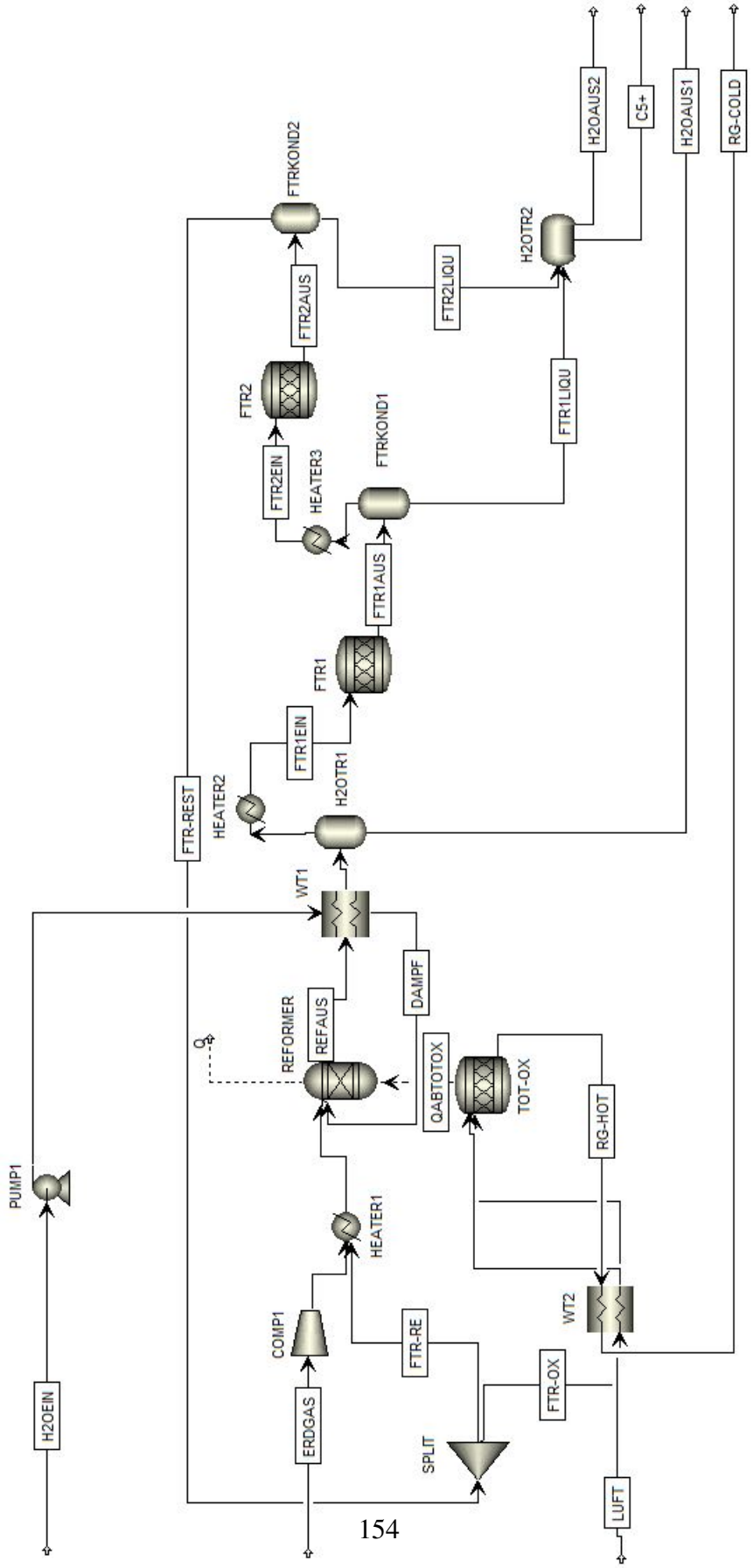

Figure D.**1:** Aspen model for the GtL process.

the FTS product flow FTR1AUS are condensed and separated as output flow FTR1LIQU in FTRKOND1. Not-converted synthesis gas is heated in HEATER3 and converted in FTR2. Subsequent, the product flow FTR2AUS is separated into a gaseous flow with alkane up to 4 C atoms (FTR-REST) and a liquid flow (FTR2LIQU). Finally, the water of both condensates (FTR1LIQU, FTR2LIQU) is separated from the C_{5+} product (output flow C5+) in decanter H2OTR2 and removed from the system (output flow H2OAUS2). This was the first operation step of the GtL process.

In the second step, the gaseous flow with alkane up to 4 C atoms (FTR-REST) is separated into two flows. One is used as input flow for the reformer (FTR-RE) and the other flow is used for the oxidation (FTR-OX). The oxidation (TOT-OX) provides energy (QABTOTOX) for the reforming. Reactants for the oxidation are pre-heated in the heat exchanger WT2. The heat from the hot oxidation products (water and carbon dioxide in RG-HOT) is used for the heat exchanger WT2. The product flow with C_1 to C_4 (FTR-RE) is added to the natural gas flow (ERDGAS) in HEATER1 and the first step starts over again until the final C_{5+} product amount is reached.

The Redlich-Kwong-Soave method (RK-Soave) is used as physical property method for calculating material flows, the NBS/NRC steam table equation of state (STEAMNBS) for the steam-water cycle in Aspen®Plus (*Trompelt*, 2009, *AspenTech*, 1999).

1.2 Design-specifications

The design-specification tool in Aspen®Plus, which correlates parameters and determines their variation, is used to achieve the desired product amount of C_{5+} with 0.01% tolerance and associated gas as variable parameter. The H_2O:C ratio of the reforming (H2OEIN) is set as varable parameter to adjust the H_2O:C ratio. In order to obtain an autothermic process the reforming process operating at 900 °C uses the released heat of the methane oxidation (TOT-OX) with a tolerance of 1kJ/mol. Defined parameters are: system pressure, chain growth probability (alpha), conversion rate, product C_{5+} amount and temperature for reforming, FTS and condenser illustrated in Table D.1.

1.3 Steam reforming

Steam reforming consists of the reforming (REFORMER) and the methane oxidation (TOT-OX) reactor. The reactor block RGibbs including phase equilibrium by using Gibbs free energy minimization is used for the reformer with H_2, CH_4, H_2O, and CO as reactants and products with $n \geq 2$ as shown in Equation D.1. The total oxidation of natural gas uses RStoic, a stoichiometric-based reactor. Despite, the molar ratio of oxygen to input flow

(FTR-OX) is set to 1.5 to reduce/avoid coking (*Baerns*, 2008).

$$C_nH_{2n+2} + (n\text{-}1)H_2O \rightarrow CH_4 + 2(n\text{-}1)H_2 + (n\text{-}1)CO \tag{D.1}$$

The reforming process operates with a 3:1 molar excessive of steam to prevent coking (*Baerns*, 2008, p.554) and to achieve the recommended H_2:CO ratio of the synthesis gas. Usually, the WGS is used to adjust the H_2 to CO ratio but in our Aspen simulation the WGS was not allowed because the H_2:CO ratio would have been around 10-20:1 due to the high amount of steam and thermodynamic constraints in Aspen. This ratio is far too high compared to the recommended ratio of 2-3:1 (*Baerns*, 2008, *Trompelt*, 2009). Finally, the H_2:CO ratio could be adjusted to 3.7:1 with a 3:1 molar excessive of steam.

1.4 FTS

RStoic is used as FTS reactor block (FTR1/2). CO is defined as variable parameter for weighting the calculated weight fractions for n = 1-30 carbon atoms according to equation D.2 for alkane. The percentage of a single product f_i to the overall reaction is calculated with the product of the selectivity W_i in wt% and the flow throughput rat CR of the FTS reactor.

$$nCO + (2n+1)H_2 \rightarrow C_nH_{2n+2} + nH_2O \tag{D.2}$$

1.5 Water separation

The temperature of the condenser (H2OTR1, FTRKOND1/2) and decanter (H2OTR2) is set to 50 °C to achieve entire water condensation and thus water separation.

1.6 Heat exchanger

The two heat exchangers WT1 and WT2 are used for heat recovering within the autothermal GtL process. WT1 recovers heat from the product flow of the reformer (REFORMER) for evaporating the water used in this reformer. WT2 works in the same way for the methane oxidation (TOT-OX). The heat of the product flow is recovered for heating the reactants of the methane oxidation.

2 Scenario parameters in Aspen

scenario no.	alpha	pressure in bar	conversion rate CR in %	temperature in °C
1	0.84	20	0.7	225
2	0.86	20	0.7	225
3	0.88	20	0.7	225
4	0.9	20	0.7	225
5	0.84	20	0.8	225
6	0.86	20	0.8	225
7	0.88	20	0.8	225
8	0.9	20	0.8	225
9	0.84	20	0.9	225
10	0.86	20	0.9	225
11	0.88	20	0.9	225
12	0.9	20	0.9	225
13	0.84	30	0.7	225
14	0.86	30	0.7	225
15	0.88	30	0.7	225
16	0.9	30	0.7	225
17	0.84	30	0.8	225
18	0.86	30	0.8	225
19	0.88	30	0.8	225
20	0.9	30	0.8	225
21	0.84	30	0.9	225
22	0.86	30	0.9	225
23	0.88	30	0.9	225
24	0.9	30	0.9	225

Table D**.1:** Scenario parameters for Aspen modeling.

APPENDIX E

Optimization

This appendix presents the Matlab source code for the GtL optimization in offshore application field by linear programming implemented by Simon Fromme.

1 GUI implemented in Matlab R2011b

```
% author: Simon Fromme, KIT 2012

function varargout = gui(varargin)
% gui MATLAB code for gui.fig
%      gui, by itself, creates a new gui or raises the existing
%      singleton*.
%
%      H = gui returns the handle to a new gui or the handle to
%      the existing singleton*.
%
%      gui('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in gui.M with the given input arguments.
%
%      gui('Property','Value',...) creates a new gui or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before gui_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to gui_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help gui

% Last Modified by GUIDE v2.5 10-May-2012 15:02:02

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @gui_OpeningFcn, ...
                   'gui_OutputFcn',  @gui_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```

```
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before gui is made visible.
% Start parameters of the optimization are getting specified here. They
% will be passed to the responsible gui elements and also saved in the
% handles structure.
function gui_OpeningFcn(hObject, eventdata, handles, varargin)

% save the default values in the handles structure for the use in other
% callback functions

%% default reactor parameters

handles.reactor_capacity_1   = 10;    %product capacity of a reactor [kg/hour]
handles.costs_operating_1    = 10;    %operating costs of a reactor [EUR/hour]
handles.utilisation_min_1    = 0.6;   %minimum capacity utilisation
handles.production_gwp_1     = 2154;   %greenhouse gas emissions
while producing the reactor [kg gwp]

handles.operating_gwp_1      = 66489;   %greenhouse gas emissions
while operating the reactor [kg gwp]

handles.production_costs_1   = 10000;  %costs for to producing the reactor [EUR]

handles.reactor_capacity_2   = 50;   %product capacity of a reactor [kg/hour]
handles.costs_operating_2    = 20;   %operating costs of a reactor [EUR/hour]
handles.utilisation_min_2    = 0.6;  %minimum capacity utilisation
handles.production_gwp_2     = 8043; %greenhouse gas emissions while producing the reactor [kg gwp]

handles.operating_gwp_2      = 302811; %greenhouse gas emissions while operating the reactor [kg gwp]

handles.production_costs_2   = 50000; %costs for to producing the reactor [EUR]

handles.reactor_capacity_3   = 100; %product capacity of a reactor [kg/hour]
handles.costs_operating_3    = 30; %operating costs of a reactor [EUR/hour]
handles.utilisation_min_3    = 0.6; %minimum capacity utilisation
handles.production_gwp_3     = 21458; %greenhouse gas emissions while producing the reactor [kg gwp]

handles.operating_gwp_3      = 632830; %greenhouse gas emissions while operating the reactor [kg gwp]

handles.production_costs_3   = 100000;  %costs for to producing the reactor [EUR]

%% years
handles.years                = 30;

%% prices

handles.price_CO2       = 0.0071;           %CO2 certificate price [EUR/kg]
handles.price_crudeOil  = 0.7311;   %crude oil price [EUR/kg]

%% other parameters
handles.product_ratio   = 0.689655172;    %product/educt ratio [dimensionless]
handles.points          = 1;    %number of dicrete points
handles.gas_gwp         = 21; %CH4 in CO2 equivalent

% pass the default parameters as a starting value to all of the individual
% gui-elements (sliders, ...)

set( handles.slider_capacity , 'Value' , handles.reactor_capacity_1 );
set( handles.slider_operatingCosts, 'Value' , handles.costs_operating_1  );
set( handles.slider_minUtilization, 'Value' , handles.utilisation_min_1 );
set( handles.slider_production_gwp, 'Value' , handles.production_gwp_1 );
set( handles.slider_operating_gwp , 'Value' , handles.operating_gwp_1 );

set( handles.slider_priceCrudeOil     , 'Value' , handles.price_crudeOil);
set( handles.slider_priceCO2          , 'Value' , handles.price_CO2     );
set( handles.slider_productEductRatio , 'Value' , handles.product_ratio );
set( handles.slider_points            , 'Value' , handles.points        );
```

```
set( handles.text_capacity        , 'String', handles.reactor_capacity_1)
set( handles.text_operatingCosts  , 'String', handles.costs_operating_1)
set( handles.text_minUtilization  , 'String', handles.utilisation_min_1)
set( handles.text_production_gwp  , 'String', handles.production_gwp_1)
set( handles.text_operating_gwp   , 'String', handles.operating_gwp_1)

set( handles.text_priceCrudeOil     , 'String', handles.price_crudeOil)
set( handles.text_priceCO2          , 'String', handles.price_CO2)
set( handles.text_productEductRatio , 'String', handles.product_ratio)
set( handles.text_points            , 'String', handles.points)

% Choose default command line output for gui
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes gui wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
% Not important for the functioning of the programm
function varargout = gui_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

%% popup menu to select the optimization method

% --- Executes on selection change in popupmenu_selectMethod.
% Doesn't do anything, since value is read directly from the element
% later on
function popupmenu_selectMethod_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
% Not important for the functioning of the programm, just the background
% color is set here
function popupmenu_selectMethod_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%% popup menu to select the reactor whose parameters are being entered

% --- Executes on selection change in popupmenu_reactorSelect.
% updates all sliders and texfields to show the values for the chosen
% reactor
function popupmenu_reactorSelect_Callback(hObject, eventdata, handles)

switch get(hObject, 'Value')
    case 1
        set(handles.slider_capacity,       'Value', handles.reactor_capacity_1);
        set(handles.text_capacity,         'String', handles.reactor_capacity_1);
        set(handles.slider_operatingCosts, 'Value', handles.costs_operating_1);
        set(handles.text_operatingCosts,   'String', handles.costs_operating_1);
        set(handles.slider_minUtilization, 'Value', handles.utilisation_min_1);
        set(handles.text_minUtilization,   'String', handles.utilisation_min_1);
        set(handles.slider_production_gwp, 'Value', handles.production_gwp_1);
        set(handles.text_production_gwp,   'String', handles.production_gwp_1);
        set(handles.text_operating_gwp,    'String', handles.operating_gwp_1);
        set(handles.slider_operating_gwp,  'Value' , handles.operating_gwp_1 );

    case 2
        set(handles.slider_capacity,       'Value', handles.reactor_capacity_2);
        set(handles.text_capacity,         'String', handles.reactor_capacity_2);
        set(handles.slider_operatingCosts, 'Value', handles.costs_operating_2);
        set(handles.text_operatingCosts,   'String', handles.costs_operating_2);
        set(handles.slider_minUtilization, 'Value', handles.utilisation_min_2);
        set(handles.text_minUtilization,   'String', handles.utilisation_min_2);
        set(handles.slider_production_gwp, 'Value', handles.production_gwp_2);
        set(handles.text_production_gwp,   'String', handles.production_gwp_2);
        set(handles.text_operating_gwp,    'String', handles.operating_gwp_2);
        set(handles.slider_operating_gwp, 'Value' , handles.operating_gwp_2 );

    case 3
```

```
        set(handles.slider_capacity,       'Value', handles.reactor_capacity_3);
        set(handles.text_capacity,         'String', handles.reactor_capacity_3);
        set(handles.slider_operatingCosts, 'Value', handles.costs_operating_3);
        set(handles.text_operatingCosts,   'String', handles.costs_operating_3);
        set(handles.slider_minUtilization, 'Value', handles.utilisation_min_3);
        set(handles.text_minUtilization,   'String', handles.utilisation_min_3);
        set(handles.slider_production_gwp, 'Value', handles.production_gwp_3);
        set(handles.text_production_gwp,   'String', handles.production_gwp_3);
        set(handles.text_operating_gwp,    'String', handles.operating_gwp_3);
        set(handles.slider_operating_gwp, 'Value' , handles.operating_gwp_3 );
end

get(hObject, 'Value')

% --- Executes during object creation, after setting all properties.
% Not important for the functioning of the programm, just the background
% color is set here
function popupmenu_reactorSelect_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%% slider and textfield to set the value for the capacity of the specified reactor

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_capacity_Callback(hObject, eventdata, handles)

switch get(handles.popupmenu_reactorSelect, 'Value')
    case 1 % 1st reactor
        handles.reactor_capacity_1 = get(hObject, 'Value')

    case 2 % 2nd reactor
        handles.reactor_capacity_2 = get(hObject, 'Value')

    case 3 % 3rd reactor
        handles.reactor_capacity_3 = get(hObject, 'Value')
end

set(handles.text_capacity, 'String', get(hObject, 'Value') );
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
% not important
function slider_capacity_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for the operating costs of the specified reactor

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_operatingCosts_Callback(hObject, eventdata, handles)

switch get(handles.popupmenu_reactorSelect, 'Value')
    case 1 % 1st reactor
        handles.costs_operating_1 = get(hObject, 'Value')

    case 2 % 2nd reactor
        handles.costs_operating_2 = get(hObject, 'Value')

    case 3 % 3rd reactor
        handles.costs_operating_3 = get(hObject, 'Value')
end
set(handles.text_operatingCosts, 'String', get(hObject, 'Value') );
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
% not important
```

```
function slider_operatingCosts_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for the minimum utilization of the specified reactor

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_minUtilization_Callback(hObject, eventdata, handles)

switch get(handles.popupmenu_reactorSelect, 'Value')
    case 1 % 1st reactor
        handles.utilisation_min_1 = get(hObject, 'Value')

    case 2 % 2nd reactor
        handles.utilisation_min_2 = get(hObject, 'Value')

    case 3 % 3rd reactor
        handles.utilisation_min_3 = get(hObject, 'Value')
end

set(handles.text_minUtilization, 'String', get(hObject, 'Value') );
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
% not important
function slider_minUtilization_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for production gwp of the specified reactor

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_production_gwp_Callback(hObject, eventdata, handles)
switch get(handles.popupmenu_reactorSelect, 'Value')
    case 1 % 1st reactor
        handles.production_gwp_1 = get(hObject, 'Value')

    case 2 % 2nd reactor
        handles.production_gwp_2 = get(hObject, 'Value')

    case 3 % 3rd reactor
        handles.production_gwp_3 = get(hObject, 'Value')
end
set(handles.text_production_gwp, 'String', get(hObject, 'Value') );
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
% not important
function slider_production_gwp_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_production_gwp (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for operating gwp of the specified reactor

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_operating_gwp_Callback(hObject, eventdata, handles)
switch get(handles.popupmenu_reactorSelect, 'Value')
    case 1 % 1st reactor
```

```
        handles.operating_gwp_1 = get(hObject, 'Value')

    case 2 % 2nd reactor
        handles.operating_gwp_2 = get(hObject, 'Value')

    case 3 % 3rd reactor
        handles.operating_gwp_3 = get(hObject, 'Value')
end
set(handles.text_operating_gwp, 'String', get(hObject, 'Value') );
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function slider_operating_gwp_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_operating_gwp (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for the crude oil Price

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_priceCrudeOil_Callback(hObject, eventdata, handles)

set(handles.text_priceCrudeOil, 'String', get(hObject, 'Value') );
handles.price_crudeOil = get(hObject, 'Value')
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function slider_priceCrudeOil_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_priceCrudeOil (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for the CO2 certificate price

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_priceCO2_Callback(hObject, eventdata, handles)

set(handles.text_priceCO2, 'String', get(hObject, 'Value') );
handles.price_CO2 = get(hObject, 'Value')
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
% not important
function slider_priceCO2_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the value for the product educt ratio

% --- Executes on slider movement.
% reads out the new value when changed, save it in the handles structure
% and writes it in the associated text field
function slider_productEductRatio_Callback(hObject, eventdata, handles)

set(handles.text_productEductRatio, 'String', get(hObject, 'Value') );
guidata(hObject, handles);handles.product_ratio = get(hObject, 'Value');

% not important
```

```
function slider_productEductRatio_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

%% slider and textfield to set the gas distribution

% Nothing is done here since the value is read directly from the slider
function popupmenu_gasFunction_Callback(hObject, eventdata, handles)

% not important
function popupmenu_gasFunction_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%% slider and textfield to set the number of discrete points used for the optimization
% --- Executes on slider movement.
function slider_points_Callback(hObject, eventdata, handles)
set(handles.text_points, 'String', get(hObject, 'Value') );
handles.points = get(hObject, 'Value');
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
% not important
function slider_points_CreateFcn(hObject, eventdata, handles)

if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on button press in pushbutton_optimize.
% This starts the actual optimization with the parameters previously
% chosen.
function pushbutton_optimize_Callback(hObject, eventdata, handles)


%reading all the parameters from the handles object
points      = get(handles.slider_points, 'Value');
gasFunction = get(handles.popupmenu_gasFunction, 'Value');

opt_method       = get( handles.popupmenu_selectMethod   , 'Value' );

%% determinig the shape of the "gas curve"
min_gasVolume = 9000000 + 2.1; %minimum amount of gas extracted during specied range
max_gasVolume = 24000000 + 2.1; %maximum amount of gas extracted during specied range

const_gasVolume = 15000000 + 0.1; %constant amount of gas extracted during specied range

%according to chosen curve type points are generated
switch gasFunction
    case 1 % flat gas curve
        gasCurve = const_gasVolume*ones(1,handles.years);
    case 2 % linear inclining gas curve
        gasCurve = linspace(min_gasVolume, max_gasVolume, handles.years);
    case 3 % linear declining has curve
        gasCurve = linspace(max_gasVolume, min_gasVolume, handles.years);
    case 4
        gasCurve = ones(1, handles.years);

        for i = 1:handles.years
         %  gasCurve(1, i) =  int64((2/points)^2*(min_gasVolume - max_gasVolume)
         % *i^2 + (4/points)*(max_gasVolume - min_gasVolume)*i + min_gasVolume);
            gasCurve(1, i) =        (2/handles.years)^2*(min_gasVolume - max_gasVolume)*i^2
            + (4/handles.years)*(max_gasVolume - min_gasVolume)*i + min_gasVolume;
        end
    % to be removed begin
    case 5
         gasCurve = [50, 100, 100]
    %to be removed end
end
```

```
gasCurve

gasCurve_split = [];
for i = 1:handles.years
    gasCurve_split = [gasCurve_split gasCurve(i)*ones(1, points)];
end
gasCurve_split

hours_per_point = 8760/points

%% According to the selected method the optimization is carried out.

switch (opt_method)

    case 1  %% revenue optimization is carried out for all three different reactor types
     by passing all relevant parameters to the external function
        [result_1 G_1] = optimizeRevenue(handles.reactor_capacity_1 * hours_per_point,
         handles.costs_operating_1 * hours_per_point, handles.utilisation_min_1, handles.price_CO2,
         handles.price_crudeOil, handles.product_ratio, handles.points*handles.years, gasCurve_split )
        [result_2 G_2] = optimizeRevenue(handles.reactor_capacity_2 * hours_per_point,
         handles.costs_operating_2 * hours_per_point, handles.utilisation_min_2, handles.price_CO2,
         handles.price_crudeOil, handles.product_ratio, handles.points*handles.years, gasCurve_split )
        [result_3 G_3] = optimizeRevenue(handles.reactor_capacity_3 * hours_per_point,
         handles.costs_operating_3 * hours_per_point, handles.utilisation_min_3, handles.price_CO2,
         handles.price_crudeOil, handles.product_ratio, handles.points*handles.years, gasCurve_split )

        result_1;
        result_2;
        result_3;

        % display the revenue in table
        set(handles.table_results, 'Visible', 'on');
        set(handles.table_results, 'data', [1 int64(G_1); 2 int64(G_2); 3 int64(G_3)]);
        set(handles.table_results, 'ColumnName',  {'reactor', 'revenue'});
        set(handles.text_results, 'String', 'results - revenue');

        set(handles.text_results, 'Visible', 'on');

    case 2 %% gwp optimization is carried out for all three different reactor types by passing all relevant parameters to the external function


        [result_1 G_1] = optimizeGWP(handles.reactor_capacity_1 * hours_per_point, handles.utilisation_min_1,
         handles.production_gwp_1, handles.gas_gwp, points * handles.years, gasCurve_split );
        [result_2 G_2] = optimizeGWP(handles.reactor_capacity_2 * hours_per_point, handles.utilisation_min_2,
         handles.production_gwp_2, handles.gas_gwp, points * handles.years, gasCurve_split );
        [result_3 G_3] = optimizeGWP(handles.reactor_capacity_3 * hours_per_point, handles.utilisation_min_3,
         handles.production_gwp_3, handles.gas_gwp, points * handles.years, gasCurve_split );

        result_1;
        result_2;
        result_3;

        % display the revenue in table
        set(handles.table_results, 'Visible', 'on');
        set(handles.table_results, 'data', [1 int64(G_1); 2 int64(G_2); 3 int64(G_3)]);
        set(handles.table_results, 'ColumnName',  {'reactor', 'gwp'});
        set(handles.text_results, 'String', 'results - gwp');
        set(handles.text_results, 'Visible', 'on');

    case 3 %% revenue2 optimization is carried out for all three different reactor types
     by passing all relevant parameters to the external function
        [result_1 G_1] = optimizeRevenue2(handles.reactor_capacity_1 * hours_per_point, handles.utilisation_min_1,
          handles.production_costs_1, handles.price_CO2, handles.price_crudeOil, handles.product_ratio,
          points * handles.years, gasCurve_split );
        [result_2 G_2] = optimizeRevenue2(handles.reactor_capacity_2 * hours_per_point, handles.utilisation_min_2,
         handles.production_costs_2, handles.price_CO2, handles.price_crudeOil, handles.product_ratio,
         points * handles.years, gasCurve_split );
        [result_3 G_3] = optimizeRevenue2(handles.reactor_capacity_3 * hours_per_point, handles.utilisation_min_3,
         handles.production_costs_3, handles.price_CO2, handles.price_crudeOil, handles.product_ratio,
         points * handles.years, gasCurve_split );

        result_1;
        result_2;
        result_3;
```

```
        % display the revenue in table
        set(handles.table_results, 'Visible', 'on');
        set(handles.table_results, 'data', [1 int64(G_1); 2 int64(G_2); 3 int64(G_3)]);
        set(handles.table_results, 'ColumnName',  {'reactor', 'gwp'});
        set(handles.text_results, 'String', 'results - gwp');
        set(handles.text_results, 'Visible', 'on');

    case 4 %% gwp optimization (production gwp + operating gwp) is carried out for all three different reactor types
     by passing all relevant parameters to the external function
        [result_1 G_1] = optimizeGWP2(handles.reactor_capacity_1 * hours_per_point, handles.utilisation_min_1,
         handles.production_gwp_1, handles.operating_gwp_1, handles.gas_gwp,
         points * handles.years, gasCurve_split );
        [result_2 G_2] = optimizeGWP2(handles.reactor_capacity_2 * hours_per_point, handles.utilisation_min_2,
         handles.production_gwp_2, handles.operating_gwp_2, handles.gas_gwp,
         points * handles.years, gasCurve_split );
        [result_3 G_3] = optimizeGWP2(handles.reactor_capacity_3 * hours_per_point, handles.utilisation_min_3,
         handles.production_gwp_3, handles.operating_gwp_3, handles.gas_gwp,
         points * handles.years, gasCurve_split );

        result_1;
        result_2;
        result_3;

        % display the revenue in table
        set(handles.table_results, 'Visible', 'on');
        set(handles.table_results, 'data', [1 int64(G_1); 2 int64(G_2); 3 int64(G_3)]);
        set(handles.table_results, 'ColumnName',  {'reactor', 'gwp'});
        set(handles.text_results, 'String', 'results - gwp');
        set(handles.text_results, 'Visible', 'on');
end

% display calc  ulated data in the diagram panel

%x-Axis: the discrete time steps
x = 1:(points * handles.years);

%y-Axis: the amount of raw gas used in the different reactors at each
%discret time stamp in optimal case
y = [gasCurve_split; result_1(1:(points * handles.years))'; result_2(1:(points * handles.years))';
result_3(1:(points * handles.years))'];

graph = handles.axes_display;

% Set division of the y-axis
set(graph, 'YLimMode', 'manual');
set(graph, 'YLim', [0 1.2 * max_gasVolume]);

% Displays the points together in step mode
h = stairs(x, y');

%% Set colors for each reactor
set(h(1),'MarkerFaceColor','black')     %% gas curve
set(h(2),'MarkerFaceColor','red')       %% reactor 1
set(h(3),'MarkerFaceColor','green')     %% reactor 2
set(h(4),'MarkerFaceColor','blue')      %% reactor 3
```

2 GWP optimization implemented in Matlab R2011b

```
% author: Simon Fromme, KIT 2012

% Determine the total reduction of CO2-Emissions considering both the
% Greenhouse Gas-Emissions (GWP) for reactor production and the Greenhouse Gas-Emissions/per year while
% operating the reactor

function [opt_result G] = optimizeGWP(reactor_capacity, utilisation_min, productionGWP, operatingGWP,
gasGWP, points, gasCurve )

%% target function
c_1 = -ones(1, points) * gasGWP;
c_2 = operatingGWP * ones(1, points);
c_3 = productionGWP;
c = [c_1 c_2 c_3];

%  logical index vector of integer values
yidx = logical( [zeros(points, 1); ones(points, 1); 1] );

%% constraints
%  1st. constraints group ( m_i <= gas(i))
constraints_1 = zeros(points, 2*points + 1);

for i = 1:points
    constraints_1(i,i) =  1;
end

%  2nd constraints group ( m_i - reactor_capacity * b <= 0 )
constraints_2 = zeros(points, 2*points + 1);

for i=1:points
    constraints_2(i , i) = 1;
    constraints_2(i , points + i) = -reactor_capacity;
end

%  3rd constraints group ( -m_i + reactor_capacity * utilisation_min * b_i <= 0 )
constraints_3 = zeros(points, 2*points + 1);

for i=1:points
    constraints_3(i , i) = -1;
    constraints_3(i , points + i ) =  reactor_capacity * utilisation_min;
end

%  4th constraints group ( b_i + -b <= 0 )
constraints_4 = zeros(points, 2*points + 1);

for i=1:points
    constraints_4(i , points + i) = 1;
    constraints_4(i , 2*points + 1 ) =  -1;
end

A = [constraints_1; constraints_2; constraints_3; constraints_4];

%  right side of constraints
b_1 = gasCurve'; % amount of gas, produced at each time
b_2 = zeros(points, 1);
b_3 = zeros(points, 1);
b_4 = zeros(points, 1);

b = [b_1; b_2; b_3; b_4];

%% Optimization

%  set equations constraints to null
Aeq = [];
beq = [];
ub  = [];
lb  = [];

%  actual optimization using mixed integer programming

%% formerly used solver
%result = miprog(c,A,b,Aeq, beq, lb, ub, yidx);
```

```
%% use gurobi optimization instead
model.A = sparse(A);
model.obj = c; %must be a dense vector
model.rhs = b; %right hand side vector
model.sense = '<';

%build the variable type vector (C = continous, I = Integer)
vtype = '';

for i=1:points
    vtype = [vtype 'C'];
end

for i=1:points
    vtype = [vtype 'I'];
end

vtype = [vtype 'I'];

model.vtype = vtype;
model.modelsense = 'min';
params.resultfile = 'revenue.lp';

params.outputflag = 0;

result = gurobi(model, params);
disp(result)

opt_result = result.x

%Calculate the revenue gained
G = -result.objval;

end
```

3 Cost optimization implemented in Matlab R2011b

```
% author: Simon Fromme, KIT 2012

% this functions assumes, that each of the reactors have just acquisition cost but no operating costs

function [opt_result G] = optimizeRevenue2(reactor_capacity, utilisation_min, production_cost,
price_CO2, price_crudeOil, product_ratio, points, gasCurve )

%% target function
c_1 = -ones(1, points)*(price_CO2 + price_crudeOil * product_ratio);
c_2 = zeros(1, points);
c_3 = production_cost;
c = [c_1 c_2 c_3];

%  logical index vector of integer values
yidx = logical( [zeros(points, 1); ones(points, 1); 1] );

%% constraints
%  1st. constraints group ( m_i <= gas(i))
constraints_1 = zeros(points, 2*points + 1);

for i = 1:points
    constraints_1(i,i) =  1;
end

%  2nd constraints group ( m_i - reactor_capacity * b <= 0 )
constraints_2 = zeros(points, 2*points + 1);

for i=1:points
    constraints_2(i , i) = 1;
    constraints_2(i , points + i) = -reactor_capacity;
end

%  3rd constraints group ( -m_i + reactor_capacity * utilisation_min * b_i <= 0 )
constraints_3 = zeros(points, 2*points + 1);

for i=1:points
    constraints_3(i , i) = -1;
    constraints_3(i , points + i ) =  reactor_capacity * utilisation_min;
end

%  4rt constraints group ( b_i + -b <= 0 )
constraints_4 = zeros(points, 2*points + 1);

for i=1:points
    constraints_4(i , points + i) = 1;
    constraints_4(i , 2*points + 1 ) =  -1;
end

A = [constraints_1; constraints_2; constraints_3; constraints_4];

%  right side of constraints
b_1 = gasCurve'; % amount of gas, produced at each time
b_2 = zeros(points, 1);
b_3 = zeros(points, 1);
b_4 = zeros(points, 1);

b = [b_1; b_2; b_3; b_4];

%% Optimization

%  set equations constraints to null
Aeq = [];
beq = [];
ub  = [];
lb  = [];

%  actual optimization using mixed integer programming

%% formerly used solver
%result = miprog(c,A,b,Aeq, beq, lb, ub, yidx);

%% use gurobi optimization instead
```

```
model.A = sparse(A);
model.obj = c; %must be a dense vector
model.rhs = b; %right hand side vector
model.sense = '<';

%build the variable type vector (C = continous, I = Integer)
vtype = '';

for i=1:points
    vtype = [vtype 'C'];
end

for i=1:points
    vtype = [vtype 'I'];
end

vtype = [vtype 'I'];

model.vtype = vtype;
model.modelsense = 'min';
params.resultfile = 'revenue.lp';

params.outputflag = 0;

result = gurobi(model, params);
disp(result)

opt_result = result.x;

%Calculate the revenue gained
G = -result.objval;

save -mat resultData.mat G;

end
```

BIBLIOGRAPHY

BIBLIOGRAPHY

Abele, E. (2008), *EcoDesign: Von der Theorie in die Praxis*, Springer-Verlag Berlin Heidelberg, Berlin and Heidelberg.

Abele, E. (2012), Simulation of the Energy Consumption of Machine Tools for a Specific Production Task, in *Leveraging technology for a sustainable world*, edited by D. A. Dornfeld, pp. 233–237, Springer.

Anderson, R. B. (1952), Fischer-Topsch Synthesis: Some important variables of the synthesis on iron catalyst, *Industrial Engineering and Process Development*, *44*(2), 391–397.

AspenTech (1999), Aspen Plus 10 [Computer software]: Physical Property Data - Reference Manual, Aspen Technology, Inc.

Azapagic, A. (1999a), Life cycle assessment and multiobjective optimisation, *Journal of Cleaner Production*, *7*(2), 135–143.

Azapagic, A. (1999b), The application of life cycle assessment to process optimisation, *Computers & Chemical Engineering*, *23*(10), 1509–1526.

Baerns, M. (Ed.) (2008), *Technische Chemie*, 1. ed., Wiley-VCH, Weinheim.

Baumann, H. (2009), *The hitch hikers's guide to LCA: An orientation in life cycle assessment methodology and application*, 3. ed., Studentlitteratur, Lund.

Betz, M. (2012), GaBi 5 [Computer software]: PE International AG, www.gabi-software.com.

Booch, G., J. Rumbaugh, and I. Jacobson (2004), UML: The unified modeling language user guide.

Bossel, H. (1994), *Modellbildung und Simulation*, 2. ed., Vieweg, Braunschweig.

Brandner, J. J. (2006), Microfabrication in Metals and Polymers, in *Micro process engineering*, *Advanced micro & nanosystems*, vol. 5, edited by N. Kockmann, pp. 267–320, Wiley-VCH, Weinheim.

Braun, S. (2012), Simulation and Prediction of Process-Oriented Energy Consumption of Machine Tools, in *Leveraging Technology for a Sustainable World*, edited by David A. Dornfeld and Barbara S. Linke, pp. 245–250, Springer.

CALCAS (2006), *Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability*, ENEA, CML Leiden University.

Charpentier, J.-C., V. Hessel, I. Vural Gürsel, Q. Wang, T. Noël, and J. Lang (2012), Potential Analysis of Smart Flow Processing and Micro Process Technology for Fastening Process Development: Use of Chemistry and Process Design as Intensification Fields, *Chemical Engineering & Technology*, *35*(7), 1184–1204.

Chemgaroo (2012), ChemgaPedia: FIZ Chemie Berlin, www.chemgapedia.de.

Ciroth, A., M. Srocka, and R. Gross (2012), OpenLCA 1.2.8 [Open source computer software]: GreenDelta, www.openlca.org.

Commission of the European Communities (2001), Green Paper on Integrated Product Policy: IPP: COM(2001) 68 final, http://eur-lex.europa.eu/LexUriServ/site/en/com/2001/com2001_0068en01.pdf.

Deshmukh, S. R. (2010), Scale-Up of Microchannel Reactors For Fischer-Tropsch Synthesis, *Industrial and Engineering Chemistry Research*, *49*, 10,883–10,888.

Diaz, N. (2010), Strategies for Minimum Energy Operation for Precision Machining.

DIN EN ISO 14040 (2006), Environmental management - life cycle assessment - principles and framework.

DIN EN ISO 14044 (2006), Environmental management - life cycle assessment - requirements and guidelines.

Eichhorn, L. (2011), Bearbeitungsparameter für Vollhartmetallschaftfräser: zylindrisch, Fa. Hitachi, Japan: Zerspanung in Edelstahl 1.4301.

Elvidge, C. D. (2009), A Fifteen Year Record of Global Natural Gas Flaring Derived from Satellite Data, *Energies*, *2*(3), 595–622.

European Parliament and Council (2005), Establishing a framework for the setting of ecodesign requirements for energy-using products: DIRECTIVE 2005/32/EC, *Official Journal of the European Union*, (191), 29–58.

Eversheim, W. (2005), *Integrierte Produkt- und Prozessgestaltung*, VDI, Springer, Berlin.

Flory, P. J. (1936), Molecular Size Distribution in Linear Condensation Polymers, *Journal of the American Chemical Society*, *58*, 1877–1885.

Garrido, J. M. (2009), *Object Oriented Simulation: A Modeling and Programming Perspective*, 1. ed., Springer-Verlag, New York.

Gasafi, E. (2006), *Entwicklung einer lebenswegbasierten Screening-Methode zur Entscheidungsunterstützung in frühen Phasen der Verfahrensentwicklung: Techn. Univ., Diss.–Darmstadt, 2006.*, *Schriftenreihe WAR*, vol. 179, Inst. WAR, Darmstadt.

GFRP (2004), Gas Flaring Reduction Projects: Framework for Clean Development Mechanism (CDM) Baseline Methodologies.

Goedkoop, M. (2012), SimaPro 7.3 [Computer software]: PRe Consultants bv.

Goedkoop, M., and R. Spriensma (2000), The Eco-indicator 99: A damage oriented method for life cycle impact assessment: Methodology Report, http://teclim.ufba.br/jsf/indicadores/holan%20ecoindicator%2099.pdf.

Goedkoop, M., and R. Spriensma (2001), The Eco-indicator 99: A damage oriented method for life cycle impact assessment: Methodology Annex, http://www.pre-sustainability.com.

Grave, A. d., S. I. Olsen, and H. N. Hansen (), Life Cycle Assessment of micro manufacturing process chains: Application to the microfactory concept: Department of Manufacturing Engineering and Management, Technical University of Denmark, Lyngby.

Griva, I., S. G. Nash, and A. Sofer (2009), *Linear and nonlinear optimization*, 2. ed., Society for Industrial and Applied Mathematics, Philadelphia and PA.

Grötschel, M. (2004), Lineare Optimierung: Algorithmische Diskrete Mathematik II: Skriptum zur Vorlesung im WS 2003/2004.

Guinee, J. (Ed.) (2001a), *Life cycle assessment - An operational guide to the ISO standards: Part 1 and 2: Scientific Background*, Ministry of Housing, Spatial Planning and Environment (VROM) and Centre of Environmental Science (CML), Den Haag and Leiden and Netherlands.

Guinee, J. (Ed.) (2001b), *Life cycle assessment - An operational guide to the ISO standards: Part 3: Scientific Background*, Ministry of Housing, Spatial Planning and Environment (VROM) and Centre of Environmental Science (CML), Den Haag and Leiden and Netherlands.

Guinee, J., G. Huppes, R. Heijungs, and E. van der Voet (2006), Research strategy, programmes and exemplary projects on life cycle sustainability analysis (LCSA): D 22 of WP 7, in *Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability*, edited by P. Masoni, ENEA, CML Leiden University.

Guinée, J. B. (2002), *Handbook on life cycle assessment: Operational guide to the ISO standards*, *Eco-efficiency in industry and science*, vol. 7, Kluwer Academic Publishers, Dordrecht and Boston.

Güttel, R., and T. Turek (2009a), Comparison of different reactor types for low temperature Fischer–Tropsch synthesis: A simulation study, *Chemical Engineering Science*, *64*.

Güttel, R., and T. Turek (2009b), Mikrostrukturierte Festbettreaktoren für stark exotherme Gasphasenreaktionen: Eine Machbarkeitsstudie, *Chemie Ingenieur Technik*, *81*(4), 495–500.

Güttel, R., U. Kunz, and T. Turek (2007), Reaktoren für die Fischer-Tropsch-Synthese, *Chemie Ingenieur Technik*, *79*(5), 531–543.

Hamming, R. W. (1997), *The art of doing science and engineering: Learning to learn*, Gordon and Breach, Amsterdam.

Häuslein, A. (2004), *Systemanalyse: Grundlagen, Techniken, Notierungen*, VDE-Verl., Berlin.

Hedemann, J. (1998-2012), Umberto v5.5 [Computer software]: ifu Hamburg GmbH.

Heijungs, R., et al. (2006), Scope and Scientific Framework for the CALCAS Co-ordination Action: D1 Scope Paper, in *Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability*, edited by P. Masoni, ENEA, CML Leiden University.

Heinrich, G. (2008), *Objektorientierte Systemanalyse*, 1. ed., Oldenbourg Wissenschaftsverlag GmbH, Oldenbourg.

Hessel, V. (2009), Homogeneous Reactions, in *Micro process engineering*, edited by V. Hessel, Fundamentals, Operations and Catalysts, pp. 367–390, Wiley-VCH, Weinheim.

Hessel, V. (2011), Novel process windows: Concept, proposition and evaluation methodology, and intensified superheated processing, *Chemical Engineering Science*, *66*, 1426–1448.

Hessel, V., D. Kralisch, and U. Krtschil (2008), Sustainability through green processing - Novel process windows intensify micro and milli process technologies, *Energy and Environmental Science*, *1*(4), 467–478.

Hessel, V., A. Renken, J. C. Schouten, and J.-I. Yoshida (2009), *Micro process engineering: A comprehensive handbook: Volume 1: Fundamentals, Operations and Catalysts*, Wiley-VCH, Weinheim.

Heydenrich, R. (2005), GTL technology overview: Analyst and investor site visit to Oryx GTL: http://www.sasol.com/sasol_internet/frontend/download.

Hochschorner, E. (2003), Evaluation of two simplified Life Cycle assessment methods, *The International Journal of Life Cycle Assessment*, *8*(3), 119–128.

Holleman, A. F. (2007), *Lehrbuch der anorganischen Chemie*, 102. ed., de Gruyter, Berlin.

Hübschmann, S., D. Kralisch, V. Hessel, U. Krtschil, and C. Kompter (2009), Environmentally Benign Microreaction Process Design by Accompanying (Simplified) Life Cycle Assessment, *Chemical Engineering and Technology*, *32*(11), 1757–1765.

Hübschmann, S., D. Kralisch, H. Loewe, D. Breuch, J. H. Petersen, T. Dietrich, and R. Scholz (2011), Decision support towards agile eco-design of microreaction processes by accompanying (simplified) life cycle assessment, *Green Chemistry*, *13*(7), 1694–1707.

Hungerbühler, K. (1999), *Chemische Produkte und Prozesse: Grundkonzepte zum umweltorientierten Design*, Springer, Berlin.

Iglesia, E. (1991), Transport-enhanced a-olefin readsorption pathways in Ru-catalyzed hydrocarbon synthesis, *Journal of Catalysis*, *129*(1), 238–256.

Joint Research Center (2010), International Reference Life Cycle Data System (ILCD) Handbook: General guide for Life Cycle Assessment, Luxembourg.

Jolliet, O., M. Margni, R. Charles, S. Humbert, J. Payet, G. Rebitzer, and R. Rosenbaum (2003), IMPACT 2002+: A new Life Cycle Impact Assessment Methodology, *International Journal of Life Cycle Assessment*, *6*(8), 324–330.

Kasper, M. (2000), *Mikrosystementwurf: Entwurf und Simulation von Mikrosystemen*, Springer, Berlin.

Kellens, K., Renaldi, W. Dewulf, and J. R. Duflou (2012), Environmental Impact Modeling of Discret Part Manufacturing Processes, in *Leveraging Technology for a Sustainable World*, edited by David A. Dornfeld and Barbara S. Linke, pp. 557–562, Springer.

Klöpffer, W. (2009), *Ökobilanz (LCA): Ein Leitfaden für Ausbildung und Beruf*, Wiley-VCH, Weinheim.

Kockmann, N. (Ed.) (2006), *Micro process engineering: Fundamentals, devices, fabrication, and applications*, *Advanced micro & nanosystems*, vol. 5, Wiley-VCH, Weinheim.

Kolb, G. (2009), Steam reforming, in *Micro process engineering: Volume 2: Devices Reactions and Applications*, edited by V. Hessel, pp. 421–443, Wiley-VCH, Weinheim.

Kralisch, D., and G. Kreisel (2007), Assessment of the ecological potential of microreaction technology, *Chemical Engineering Science*, *62*, 1094–1100.

Kunnari, E. (2009), Environmental evaluation of new technology: printed electronics case study, *Journal of Cleaner Production*, *17*, 791–799.

Kurz, H. D. (2009), *Von Vilfredo Pareto bis Amartya Sen*, *Beck'sche Reihe*, vol. 1859, 1 ed., Beck, München.

Lambrecht, H. (2011), Stoffstromnetzbasierte Optimalplanung des Ressourceneinsatzes industrieller Produktionssysteme, Ph.D. thesis, Leuphena Universität, Lüneburg.

Lindner, J. P. (2010), LCA at the earliest stages of process development: Frauenhofer IBP, Echterdingen.

Mathworks (2012), MATLAB R2011b: http://www.mathworks.de.

Matousek, J. (2007), *Understanding and Using Linear Programming*, Universitext, Springer, Berlin and Heidelberg.

Mills, P. L., D. J. Quiram, and J. F. Ryley (2007), Microreactor technology and process miniaturization for catalytic reactions - A perspective on recent developments and emerging technologies: 8th International Conference on Gas-Liquid and Gas-Liquid-Solid Reactor Engineering, *Chemical Engineering Science*, *62*(24), 6992–7010.

Mitropetros, K., and A. Bazzanella (2010), *Leitfaden Mikroverfahrenstechnik*, DECHEMA, Frankfurt am Main.

Müller, K. G., M. U. Lamperth, and F. Kimura (2004), Parameterised Inventories for Life Cycle Assessment: Systematically Relating Design Parameters to the Life Cycle Inventory, *International Journal of LCA*, *9*(4), 227–235.

Myrstad, R., S. Eri, P. Pfeifer, E. Rytter, and A. Holmen (2009), Fischer-Tropsch synthesis in a microstructured reactor, *Catalysis Today*, *147*, 301–304.

National Geophysical Data Center (NOAA) (2011), *Global Gas Flaring Estimates*, http://www.ngdc.noaa.gov/dmsp/interest/gas_flares.html.

Niemeyer, G. (1977), *Kybernetische System- und Modelltheorie: System dynamics*, Systemstudium Wirtschaftsinformatik, Vahlen, München.

Ophey, L. (2005), *Entwicklungsmanagement: Methoden in der Produktentwicklung*, VDI, Springer, Berlin.

Overcash, M. (2012), Unit Process Life Cycle Inventory (UPLCI): A Structured Framework to Complete Product Life Cycle Studies, in *Leveraging Technology for a Sustainable World*, edited by David A. Dornfeld and Barbara S. Linke, pp. 1–4, Springer.

Park, P.-J., K.-M. Lee, and W. Wimmer (2006), Development of an Environmental Assessment Method for Consumer Electronics by combining Top-down and Bottom-up Approaches (11 pp), *The International Journal of Life Cycle Assessment*, *11*(4), 254–264.

Paucksch, E., S. Holsten, M. Linß, and F. Tikal (2008), *Zerspantechnik: Prozesse, Werkzeuge, Technologien*, Studium, 12. ed., Vieweg + Teubner, Wiesbaden.

Peters, M. S., K. D. Timmerhaus, and R. E. West (2003), *Plant design and economics for chemical engineers*, McGraw-Hill chemical engineering series, 5. ed., McGraw-Hill, Boston.

Pfeifer, P. (2009), Partial Oxidation, in *Micro process engineering: Volume 2: Devices Reactions and Applications*, edited by V. Hessel, pp. 445–464, Wiley-VCH, Weinheim.

Pfeifer, P. (2012), *Application of Catalysts to Metal Microreactor Systems*, Vivek Patel, Chemical Kinetics.

Puskas, I. (2003), Comments about the causes of deviations from the Anderson–Schulz–Flory distribution of the Fischer–Tropsch reaction products, *Catalysis Today*, *84*, 99–109.

Rennard, D., R. French, S. Czernik, T. Joephson, and L. Schmidt (2010), Production of synthesis gas by partial oxidation and steam reforming of biomass pyrolysis oils, *International Journal of Hydrogen Energy*, *35*, 4048–4059.

Ropohl, G. (1979), *Allgemeine Technologie: Eine Systemtheorie der Technik*, 1. ed., Carl Hanser Verlag, München and Wien.

Ropohl, G. (2009), *Allgemeine Technologie: Eine Systemtheorie der Technik: Univ., Habil.-Schr.–Karlsruhe, 1978.*, 3., überarb. aufl. ed., Univ.-Verl. Karlsruhe and Univ.-Bibl., Karlsruhe.

Rydh, C. J. (2005), Life cycle inventory data for materials grouped according to environmental and material properties: Life Cycle Assessment, *Journal of Cleaner Production*, *13*(13-14), 1258–1268.

Schepelmann, P., M. Ritthof, H. Jeswani, A. Azapagic, and K. Suomalainen (2006), Options for deepened and broadened LCA: Deliverable (D17) of work package 3 (WP3) CALCAS project, in *Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability*, edited by P. Masoni, ENEA, CML Leiden University.

Schirrmeister, S. (2006), Design Process and Project Management, in *Micro process engineering*, *Advanced micro & nanosystems*, vol. 5, edited by N. Kockmann, pp. 203–234, Wiley-VCH, Weinheim.

Schmidt, M. (2007), AP 3 - Parametrisierung: Projektbericht: im Rahmen des Forschungsvorhabens FKZ 01 RN 0401 im Auftrag des Bundesministeriums für Bildung und Forschung, Karlsruhe.

Schmidt, M. (2009), *Stoffstrombasierte Optimierung: Wissenschaftliche und methodische Grundlagen sowie softwaretechnische Umsetzung*, MV-Wissenschaft, Verl.-Haus Monsenstein und Vannerdat, Münster.

Schrems, S., C. Eisele, and E. Abele (2011), Methodology for Energy and Resource Efficient Process Chain Design, in *Proceedings of 18th CIRP International Conference on Life Cycle Engineering*, edited by J. Hesselbach, pp. 299–304, Springer.

Schulz, H. (1999), Kinetic modelling of Fischer–Tropsch product distributions, *Applied Catalysis A*, *General 186*, 91–107.

Stachowiak, H. (1973), *Allgemeine Modelltheorie*, Springer, Wien.

Steynberg, A. (2006), *Fischer-Tropsch technology*, *Studies in surface science and catalysis*, vol. 152, reprinted. ed., Elsevier, Amsterdam.

Tavakoli, A. (2008), Application of Anderson–Schulz–Flory (ASF) equation in the product distribution of slurry phase FT synthesis with nanosized iron catalysts, *Chemical Engineering Journal*, *136*, 358–363.

The World Bank (2002), Global Gas Flaring Reduction (GGFR): A Public-Private Partnership, Johannesburg.

Tonkovich, A. L. (2004), Microchannel process technology for compact methane steam reforming, *Chemical Engineering Science*, *59*, 4819–4824.

Tonkovich, A. L. (2011), Microchannel Gas-to-Liquids for Monetizing Associated and Stranded Gas Reserves, *Oxford Catalyst Group*, http://www.velocys.com/press/wp.

Trompelt, M. (2009), Modellierung der Fischer-Tropsch-Synthese mit Aspen Plus, *Erdöl, Erdgas, Kohle*, *125*(2), 79–84.

UNCED (1999), *Agenda 21: Programme of action for sustainable development: Rio declaration on environment and development*, reprint. ed., United Nations, Geneva.

United Nations (1995), Climate Change: The Science of Climate Change: Summary for Policymakers and Technical Summary of the Working Group 1.

van Bertalanffy, L. (2001), *General system theory: Foundations, development, applications*, 13. ed., Braziller, New York.

van Vliet, O. P. R. (2009), Fischer-Tropsch diesel production in a well-to-wheel perspective: a carbon, energy flow and cost analysis, *Energy Conversion and Management*, *50*, 855–876.

Vester, F. (2007), *The art of interconnected thinking: Ideas and tools for a new approach to tackling complexity*, 6. ed., MCB-Verl., München.

Vester, F. (2011), *Die Kunst vernetzt zu denken: Ideen und Werkzeuge für einen neuen Umgang mit Komplexität*, *Dtv*, vol. 33077, 8. ed., Deutscher Taschenbuch-Verlag, München.

Weidema, B., and R. Hischier (2010), Ecoinvent v2.2 [Computer software]: Ecoinvent Centre c/o EMPA St. Gallen.

Yuan, C. Y., and D. A. Dornfeld (2010), Integrated Sustainability Analysis of Atomic Layer Deposition for Microelectronics Manufacturing, *Journal of Manufacturing Science and Engineering*, *132*(3).

Zamagni, A. (2006), Critical review of the current research needs and limitations related to ISO-LCA practice: D7 of WP 7, in *Co-ordination Action for innovation in Life-Cycle Analysis for Sustainability*, edited by P. Masoni, ENEA, CML Leiden University.

Zschieschang, E., P. Pfeifer, and L. Schebek (2012a), Environmentally optimized microreactor design through Life Cycle Assessment, *Green Processing and Synthesis*, *1*, 375–384.

Zschieschang, E., P. Pfeifer, and L. Schebek (2012b), Modular Server - Client - Server (MSCS) Approach for Process Optimization in Early R&D of Emerging Technologies by LCA, in *Leveraging Technology for a Sustainable World*, edited by David A. Dornfeld and Barbara S. Linke, pp. 119–124, Springer.

Eva Zschieschang
geb. in Dresden

Wissenschaftlicher Werdegang

2013	Promotion an der Technischen Universität Darmstadt am Institut IWAR, Fachgebiet Industrielle Stoffkreisläufe bei Frau Prof. Liselotte Schebek mit dem Titel *„Life Cycle Assessment in Technology Development - The Case of Micro Process Engineering"*.
2008	Diplomarbeit am Fraunhofer Institut für Solare Energiesysteme ISE in Freiburg zum Thema: *„Entwicklung eines neuen Zellkonzepts für ein PEM-Mikrobrennstoffzellensystem"*.
2003 - 2008	Studium der Mikrosystemtechnik an der Universität Freiburg mit Vertiefung Systemtechnik (Schwerpunkt: Mikro- & Nanooptik, Nanotechnologie, Erneuerbare Energien)
2000 - 2002	Ausbildung zur technischen Assistentin für Biologie und Chemie

Beruflicher Werdegang

seit Dez. 2012	Wissenschaftliche Mitarbeiterin am Institut für Industrial Ecology (INEC), Hochschule Pforzheim, Pforzheim
2009 - 2012	Wissenschaftliche Mitarbeiterin am Institut für Technikfolgenabschätzung und Systemanalyse (ITAS) am Karlsruher Institut für Technologie (KIT), Karlsruhe
2008	Wissenschaftliche Mitarbeiterin in der Abteilung von Dr. Christopher Hebling (Wasserstofftechnologie, Mikroenergietechnik) am Fraunhofer Institut für Solare Energiesysteme (ISE), Freiburg
2002 - 2003	Technische Assistentin in der Arbeitsgruppe von Dr. Takashi Hiiragi am Max Planck-Institut für Immunbiologie und Epigenetik (MPI), Freiburg